ADVANCED CFD METHODS FOR WIND TURBINE ANALYSIS

A Dissertation
Presented to
The Academic Faculty

by

C. Eric Lynch

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Aerospace Engineering

Georgia Institute of Technology
May 2011

ADVANCED CFD METHODS FOR WIND TURBINE ANALYSIS

Approved by:

Professor Marilyn Smith,
Committee Chair
School of Aerospace Engineering
Georgia Institute of Technology

Professor Olivier Bauchau
School of Aerospace Engineering
Georgia Institute of Technology

Professor Narayanan Komerath
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Sam Shelton
Strategic Energy Institute
Georgia Institute of Technology

Dr. Christopher Stone
Computational Science & Engineering, LLC

Date Approved: January 18, 2011

Dedicated to my wife, Elizabeth.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my advisor, Dr. Marilyn Smith for her guidance through my years in graduate school. Thanks also go to the other members of my committee for their helpful suggestions.

This work was supported in part by the National Science Foundation, Project 0731034, "Advancing Wind Turbine Analysis and Design for Sustainable Energy". I would like to thank NSF and the NSF Program Officers, Dr. Trung Van Nguyen and Dr. Greg Rorrer, for their support in this endeavor. Any opinions, findings, and conclusions or recommendations expressed in this material do not necessarily reflect the views of the National Science Foundation. Computational support for this research was supported in part by the National Science Foundation through TeraGrid[21] (grant # TG-MSS080017N). TeraGrid resources were hosted by NCSA, LONI, and TACC. Sincere thanks go Dr. Christopher Stone for his assistance in obtaining Teragrid time.

I would like to thank Bob Biedron and Beth Lee-Rausch at NASA Langley for providing the HART grids used in this work as well as the data from their HART simulations for comparison. Thanks also go to Scott Schreck at NREL for providing the NREL UAE Phase VI datasets.

I would like to thank the members of the Fun3d development team for allowing me access to the Fun3d SVN repository so I could share the HRLES turbulence model and kd-tree nearest neighbor search with other users. I also thank Ralph Noack of the Penn State ARL for all his assistance using Suggar++ (and putting up with my bug reports).

I offer my sincere thanks to the students of the Computational Aeroelasticity Lab, both past and present, for all their help over the years. In particular, I would like to thank Jennifer Abras and Nic Reveles for numerous helpful discussions about CFD/CSD coupling. Thanks also go to Rajiv Shenoy for his help with grid generation and Jean de Montaudouin for building Dymore decks.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Horizontal-axis wind turbines operate in a complex, inherently unsteady aerodynamic environment. Even when the rotor is not stalled, the flow over the blades is dominated by three-dimensional (3-D) effects. Stall is accompanied by massive flow separation and vortex shedding over the suction surface of the blades. Under yawed conditions, dynamic stall may be present as well. In all operating conditions, there is bluff-body shedding from the turbine nacelle and support structure which interacts with the rotor wake. In addition, the high aspect ratios of wind turbine blades make them very flexible, leading to substantial aeroelastic deformation of the blades, altering the aerodynamics. Finally, when situated in a wind farm, turbines must operate in the unsteady wake of upstream neighbors.

Though computational fluid dynamics (CFD) has made significant inroads as a research tool, simple, inexpensive methods, such as blade element momentum (BEM) theory, are still the workhorses in wind turbine design and aeroelasticity applications. These methods generally assume a quasi-steady flowfield and use two-dimensional aerodynamic approximations with very limited empirical 3-D corrections. As a result, they are unable to accurately predict rotor loads near the edges of the operating envelope. CFD methods make very few limiting assumptions about the flowfield, and thus have much greater potential for predicting these flows. In this work, a range of unstructured grid CFD techniques for predicting wind turbine loads and aeroelasticity has been developed and applied to a wind turbine configuration of interest.

First, a nearest neighbor search algorithm based on a k-dimensional tree data structure was used to improve the computational efficiency of an approximate unsteady actuator blade method. This method was then shown to predict root and tip vortex locations and strengths similar to an overset method on the same background mesh, but without the computational expense of modeling the blade surfaces. A hybrid Reynolds-averaged Navier-Stokes / Large Eddy Simulation (HRLES) turbulence model, previously developed for structured grids,

was extended to an unstructured framework. It was demonstrated to improve predictions of unsteady loading and shedding frequency in massively separated cases. The sensitivity of the model to highly stretched grid topologies was also explored. For aeroelastic predictions, a methodology for tight coupling between an unstructured CFD solver and a computational structural dynamics tool was developed. Due to the lack of experimental data pertaining to a flexible turbine, the coupling algorithm was validated for a helicopter rotor, but the method is sufficiently general that it can be immediately applied to a wind turbine when suitable correlation data becomes available in the future. Finally, time-accurate overset rotor simulations of a complete turbine—blades, nacelle, and tower—were conducted using both RANS and HRLES turbulence models. The HRLES model was able to accurately predict rotor loads when stalled. In yawed flow, excellent correlations of mean blade loads with experimental data were obtained across the span, and wake asymmetry and unsteadiness were also well-predicted.

CHAPTER I

MOTIVATION

The United States is in need of alternative energy sources to replace current hydrocarbon sources, including coal, natural gas, and petroleum. These resources are non-renewable and emit greenhouse gases into the atmosphere. In the case of petroleum, a large portion of the global supply comes from geopolitically unstable regions of the world. It is estimated that the United States has sufficient wind resources to provide up to 20% of its energy [1]. Figure 1 shows a map of wind energy availability in the United States. There are currently large stretches of land with high wind availability that are grossly underutilized. However, there are a number of challenges, both technical and political, that must be overcome before wind energy can compete with other power generation methods.

In order to be a competitive energy resource, wind turbines must be efficient in terms of power generated per dollar when compared to more traditional resources. The torque output of a wind turbine, which is directly related to the power output, scales linearly with the swept area of the rotor disc. Therefore, to increase the total generating capacity of a wind farm, either larger turbine rotors or a larger number of turbines are needed. Since such a large percentage of the total cost of a turbine is absorbed by the support structure (i.e. the tower) and other parts that do not directly contribute to energy capture, a smaller number of turbines with larger rotors is the most cost-effective strategy. On a modern utility-scale turbine, rotor diameter can be on the order of 100 meters. As blade length increases, aeroelastic effects become more important, contributing to fatigue that shortens the life of a turbine. Furthermore, building blade and tower structures capable of withstanding those loads increases weight and therefore cost of the turbine.

To better carry the structural loads of large turbines, blades with very thick airfoil sections have been proposed. The effect of these thick sections on aerodynamic performance

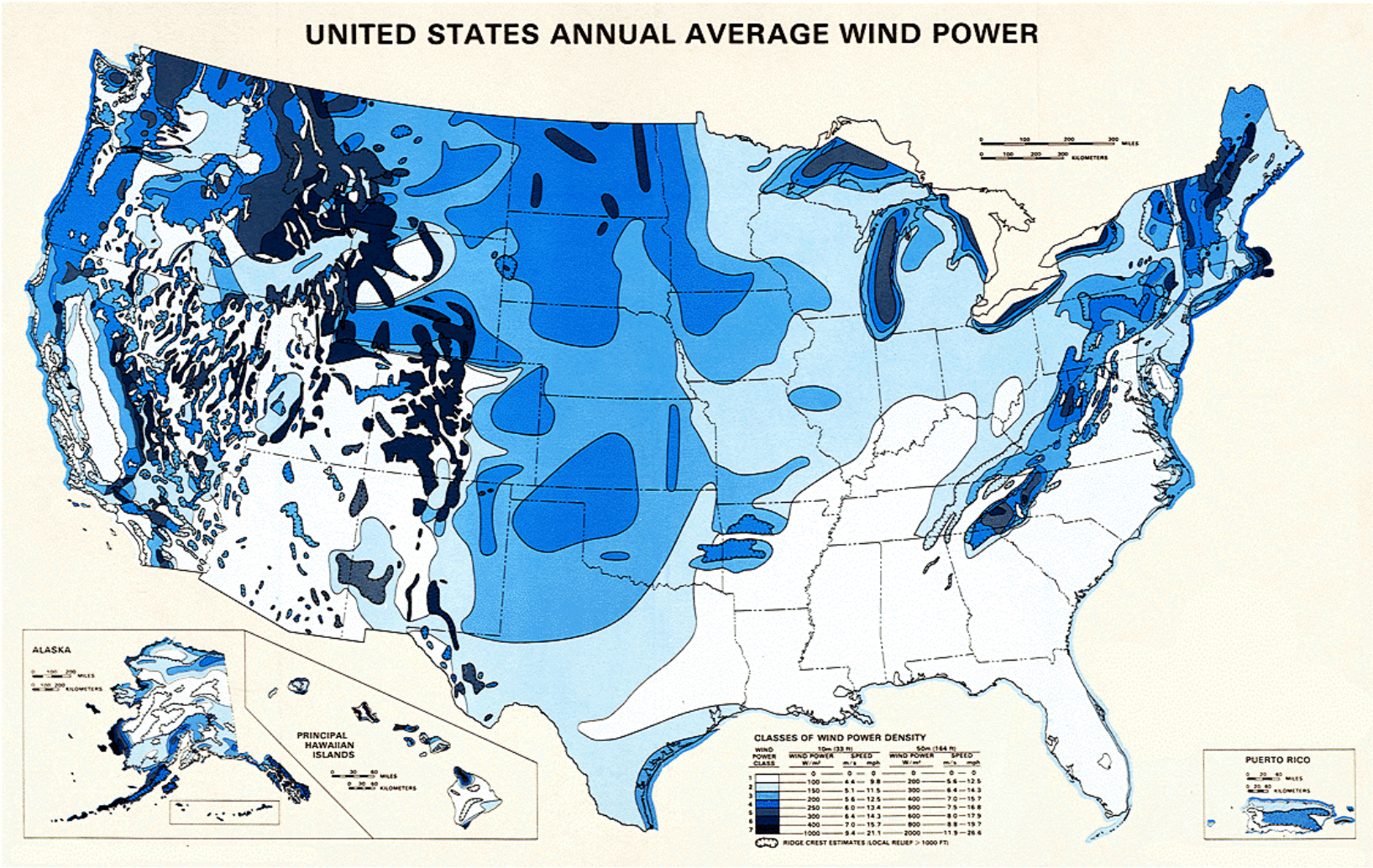


Figure 1: United States annual average wind power [31].

is difficult to predict using current tools, as they rely heavily on blade element momentum theory (BEMT), which cannot accurately predict the response of blades that violate thin airfoil assumptions. Furthermore, such methods model aerodynamics as basically two-dimensional (2-D) phenomena, yet very large turbines with blunt airfoil sections exhibit a great deal of three-dimensioanl (3-D) behavior in their flowfields. Analysis of such complex flowfields requires a more physics-based approach, such as that provided by computational fluid dynamics (CFD). Unlike BEMT methods, which neglect 3-D effects and aerodynamic phenomena occurring off the blade surfaces, CFD models the complete aerodynamic environment using physics-based, rather than empirical, equations.

Aeroacoustic noise is a major barrier to greater public acceptance of wind turbines in the energy infrastructure. On a turbine, the major sources of broadband noise are inflow turbulence, interaction of the turbulent boundary layer with the airfoil trailing edge, and vortex shedding noise due to the finite thickness of the trailing edge [34]. However, there is also very low frequency noise associated with vortex shedding from the tower [49], typically on the order of 20 Hz. Low frequency noise is attenuated by the atmosphere less than the higher frequency broadband noise from the blades. In a downwind turbine configuration, this is only magnified by the interaction of the blades with the turbulent tower wake. Recently, there have been reports of numerous health problems for people living close (less than 1 km) to a wind farm, due to constant low frequency noise [24]. Aeroacoustic prediction methods currently used in the wind industry tend to be empirical in nature, relying on assumptions that may be incorrect for modern designs. Modern aeroacoustic methods that use CFD flowfield data as their input stand to greatly improve acoustic predictions since CFD proceeds directly from the underlying equations of fluid motion and relies on fewer assumptions that limit its applicability. Since noise comes from infinitesimal pressure fluctuations with magnitudes on the same order as error terms in CFD solutions, accurate aeroacoustic predictions of this type require flowfield characterizations with accuracy that can only be reached when aeroelastic effects are included.

CHAPTER II

BACKGROUND

Despite their relatively simple appearance, wind turbines operate in an aerodynamic environment that is challenging to numerically model. Due to the atmospheric boundary layer, there can be considerable variation in wind speed between the top and bottom of the rotor disc. If the turbine is not facing directly into the wind, i.e. if there is yaw, the blade loads will vary cyclically as they rotate. All wind turbines have some kind of yaw control, but the wind direction can vary too quickly for the controller to maintain zero yaw. At high wind speeds, even in axial flow, the blades may be stalled.

To date, most computational studies of wind turbine aerodynamics have focused on simple, inexpensive methods based on blade element momentum (BEM) theory. These methods provide basic insight into turbine flows, but only under the simplest conditions: constant wind speed with zero yaw. BEM methods operate under what Leishman calls the *independence principle*, in which the aerodynamics of each airfoil section along the blades are computed independently of neighboring sections [45]. As a result BEM methods completely neglect spanwise flow and other 3-D effects, which have been shown to result in significant lift augmentation and stall delay, especially near the blade roots [83]. As a result, traditional BEM methods, even when they incorporate some sort of 3-D correction, typically under-predict torque. Designs based on such simulations can result in structures that succumb to fatigue sooner that expected [36, 25].

Computational fluid dynamics techniques can mitigate many of the inaccurate simplifying assumptions used in common wind turbine analysis methods. However, as will be discussed shortly, CFD analyses have their own shortcomings. What follows is four broad classes of CFD techniques that can offer improvements over current wind turbine design tools. The first is actuator blade methods, which can provide physics-based characterizations of wind turbine wakes while reducing computational expense associated with modeling

the blades. The second is hybrid Reynolds Averaged Navier-Stokes / Large Eddy Simuation (RANS/LES) methods, which can improve the prediction of unsteady, separated flows. The third is overset methods, which introduce the ability to treat the relative motion between rotor and its support structure. The final technique is coupling between CFD and a computational structural dynamics methodology to model the aeroelastic response of the rotor blades.

2.1 Actuator Disc and Blade Techniques

Actuator disc techniques are one approach to managing complexity and computational expense in simulations of wind turbines. Actuator disc methods are based on momentum theory and seek to model the *effects* of a rotor on the surrounding flowfield without modeling the rotor itself. Since the flow over the rotor blades produces lift and drag forces on the blades, there must be another force equal in magnitude and opposite in direction acting on the flow. This reaction force can be included in the underlying CFD model in two ways. It can be implemented like a special boundary condition, where a permeable boundary surface is embedded in the mesh, with the flow velocity constrained to be continuous through the surface while the pressure is discontinuous. Alternatively, those forces can be included as body forces in the Navier-Stokes equations. The body force approach offers additional flexibility over the pressure discontinuity. Since no surface is needed in the grid, the position of the disc can change, allowing different choices of the tip path plane on the same grid. In either case, the blades themselves are neglected. The pressure discontinuity or body forces provide the same effects on the flowfield as rotating blades, but in a time-averaged sense. Without modeling the blades directly, CFD simulations can either use fewer grid points, speeding their execution compared to more detailed methods, or place more points in the turbine wake gaining greater modeling fidelity.

CFD simulations that employ actuator discs has seen widespread use in both the rotorcraft and wind industries. For a good review of actuator disc techniques, largely from the perspective of structured grid CFD, see Reference [42]. O'Brien implemented unsteady actuator disc and and blade techniques in the unstructured solver FUN3D and compared

those against fully overset techniques on several geometries, finding that an actuator disc dramatically improved predictions of helicopter fuselage loads [68, 67]. Schweikhard [86] implemented a steady-state actuator disc in another unstructured framework that was very similar to that of O'Brien except that the boundary surface approach was used. While this approach fixes the location of the rotor relative to other bodies in the flow, the use of boundary surfaces allows for flat boundary layer-like cells to be used at the disc to capture the strong gradients normal to the disc without excessively increasing node count in the radial direction.

While actuator disc methods provide a good approximation of the influence of the rotor, the fact that they lack discrete blades means that they model the rotor in an azimuthally averaged fashion. This makes them unsuitable for cases with yaw error since the rotor wake varies azimuthally, not just radially. Furthermore, though O'Brien showed that they can capture the roll-up of two wing-like vortices in the near wake, they cannot capture discrete tip vortices [67]. To work around this problem, Sørensen et al. [96] used an actuator-line method, whereby the blades are modeled by lines along which body forces act. These body forces are typically derived from a BEM method that extracts aerodynamic response from tabulated airfoil data. In this manner, the unsteady effect on the flowfield by individual blades can be included in analysis while still avoiding direct CFD modeling of the blade surfaces and their associated boundary layers. This actuator line method was applied to a 500 kW Nordtank turbine and achieved good agreement with the experimental power curve for pre-stall wind speeds. Above about 12 m/s, where that particular rotor is stalled, the method over-predicted power because the airfoil data, lacking three-dimensional effects, is less accurate in that regime.

The actuator line method has also been applied with good results by Mikkelsen [58], who compared it against an axisymmetric actuator disc approach and used it to validate some of the fundamental assumptions of traditional BEM methods. Whereas Sørensen et al. [96] used a code that solves the Navier-Stokes equations in vorticity-velocity form, Mikkelsen used the incompressible code EllipSys3D, which solves in pressure-velocity form. This decision allows for the inclusion of solid boundaries like the turbine tower.

The EllipSys3D code was also used by Ivanell et al. [38], who evaluated the vortical wake structures produced by the actuator line method. In that work, the actuator lines were fixed in the grid, and the effects of blade rotation were applied via a boundary condition, which allowed them to use an efficient steady-state formulation. While this significantly decreases computational expense, it makes the method unsuitable for yawed cases. In addition, the azimuthal boundary condition was also periodic, so a three-bladed rotor could be modeled with a grid covering a 120° sector of the rotor disc. Again, this precludes the method being applied to any case that does not have a periodic solution, exactly the cases that should benefit the most from a detailed CFD approach.

In addition to actuator discs, O'Brien also developed an actuator method that lay between the actuator disc and line concept. This method, termed actuator "blades," was evaluated on helicopter rotor-fuselage interaction problems [68, 67]. The key difference between actuator line and actuator blade methods is that in the former, each blade consists of a single line of sources. To avoid discontinuities, each source's loading is distributed over multiple grid points using a "regularization function" that makes a source's influence at a distance r away from it scale with $\exp(r^2)$. On the other hand, the actuator blade method of O'Brien uses a rectangular array of sources for each blade, providing a continuous influence without a regularization function. This should provide a more accurate local influence since the local angle of attack (a key input to the underlying BEM model) can vary with chord.

Though they are undoubtedly easier to apply to complex geometries, unstructured actuator methods like that of O'Brien [67] suffer from one drawback besides the usual computational overhead associated with unstructured grids. In order to apply body forces from the actuator sources to the flow, it is necessary to know which grid node (or cell centroid for a cell-centered code) is closest to each actuator source, which is continuously in motion with respect to the inertial grid. This entails some sort of nearest neighbor search procedure. In actuator disc cases or actuator line/blade cases in which the sources are fixed in the grid (perhaps because the equations are being solved in a rotating frame), this search can be performed as a one-time pre-processing step. In that case, any search algorithm will be sufficient. However, if the sources move, the search must be repeated at each time step.

In the structured grid actuator methods discussed so far, the grid has a regular structure in memory that mimics the structure in space, allowing one to efficiently search through a range of grid indices. In an unstructured grid, a much more general search procedure is required. In O'Brien's FUN3D formulation, this search is done in an exhaustive brute-force manner, by looping over all the grid nodes to find the node associated with a single source, and then repeating that loop for all the other sources.

Numerous other approximate techniques exist for predicting rotor flows, including vortex lattice and vortex particle methods [20, 85]. These methods have the advantage of not requiring a large volume mesh to discretize the flowfield and record the disturbances created by bodies within them. Instead, the bodies shed particles that carry vorticity into the wake. However, they often rely on idealized models of fluid flow that prevent them from being used to predict flowfield features dominated by viscous effects, such as separation.

2.2 Hybrid RANS/LES

Most fluid flows of engineering interest are turbulent, and wind turbines are no exception. While numerous advances have been made in the numerical solution of the Navier-Stokes equations, turbulent flows still present challenges for today's methods. Turbulent flows are characterized by a very wide range of scales in both time and space. Most of the kinetic energy of a turbulent flow is stored in the large-scale structures of the flow. In contrast, kinetic energy is dissipated as heat at the smallest scales. The size of the smallest length scales, and therefore the maximum allowable CFD grid spacing to completely resolve all turbulence, is inversely proportional to $Re^{3/4}$. So for a 3-D simulation, the number of grid points must scale with $Re^{9/4}$ [27]. Because such fine grids and small timesteps are not practical with today's computers, compromises must be made that approximate certain aspects of the physics.

Most CFD methods solve the Reynolds-averaged Navier-Stokes (RANS) equations. Essentially, only the mean flow is solved on the computational mesh, and the turbulent physics are replaced by *closure* models of varying sophistication. RANS methods include simple

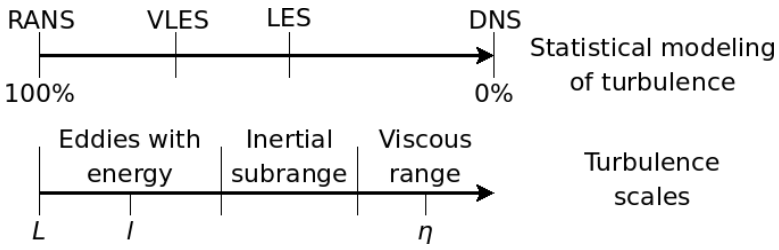


Figure 2: Illustration of the resolution of turbulence scales by numerical techniques.

algebraic models like Baldwin-Lomax, one-equation models like Baldwin-Barth and Spalart-Allmaras, and two-equation models such as k-ω and k-ω SST. These turbulence models tend to be heuristic and rely on non-physical constants that are "tuned" to specific flows, such as airfoils at low angles of attack. That is, the results from the model are iteratively compared against experimental data, and the constants are adjusted until the computational results agree with the experiment. While they perform well for those cases, they fail to accurately predict unsteady flows dominated by viscous effects, as in the case of static and dynamic stall [93, 107] or bluff-body flows.

Large-eddy simulation (LES) is a more physically accurate CFD technique than RANS, in which the governing equations are mass-averaged and then filtered such that turbulence eddies larger than the grid size are captured and turbulence with scales smaller than the grid is still modeled. As Davidson notes [27], the rationale for this is that very little energy actually cascades from the smaller scales back into the large scales, so approximations made at the small scales should have few effects on the large scale flow features that are of engineering interest. Strictly speaking, LES requires that the resolved scales extend into the inertial sub-range. These scales, and those smaller, are more *universal*, meaning that their physics depends less on the particular geometry. Such universal scales are more amenable to modeling, since even when heuristics are used, they should be valid for a larger range of flows. The drawback of LES is that in order to directly solve for the larger turbulent scales, it requires much finer grids and smaller time steps than unsteady RANS (URANS). In most engineering applications, LES remains far too computationally expensive for routine use.

Hybrid RANS/LES methods provide a way to achieve some of the advantages of LES for separated and highly vortical flowfields while retaining the computational efficiency of

RANS. Baurle et al. [12] developed the idea that RANS and LES methods can be linearly blended by some smooth function to form a hybrid model. This idea of blending was utilized in 2006 by Sanchez-Rocha et al. [79], who used the k-ω SST RANS model as the basis for a hybrid method within an existing LES code to resolve wall-bounded turbulence. Sanchez-Rocha demonstrated this capability with simulations of a NACA 0015 airfoil in static and dynamic stall at a Reynolds number of 1 million. A more common approach is to incorporate an LES model into a RANS code to capture subgrid scales, so that where grid resolution permits, LES-like results are obtained. On coarse grids that are usually only suitable for RANS applications, hybrid methods of this variety can capture larger turbulent eddies in the interior of the flow (away from boundaries), thus providing a better approximation of the true physics than RANS alone.

These two approaches to hybrid RANS-LES, are different in the sense of the underlying methodologies. LES codes are often explicit, with high order spatial and temporal schemes. Care is taken to include most, if not all, of the fluctuating terms that appear when mass-averaging the compressible Navier-Stokes equations. Within existing CFD (URANS) codes, the spatial and temporal schemes are usually of lower order than LES codes. This is especially true of unstructured codes, where spatial accuracy beyond second order is very difficult to implement. The temporal integration is typically implicit to allow larger time steps while maintaining the stability of the numerical scheme. Fluctuating terms appearing from the time-averaging of the incompressible or compressible Navier-Stokes equations may be simplified so that they are less computationally expensive. Finally, the schemes may be overly dissipative, resulting in a smearing of the features of the flow field.

Detached Eddy Simulation (DES) is a common hybrid formulation, in which the turbulence near walls is modeled with Menter's k-ω SST [54] or Spalart-Allmaras (SA) [98] RANS turbulence models. However, it must be noted that DES models are separate and distinct from hybrid methods like those of Sanchez-Rocha in that they lack a dedicated subgrid-scale model. Instead, the RANS equations perform "double-duty" as the LES model by modifying a length scale used in the destruction terms [109, 106]. In the baseline SA model, distance from the nearest wall is used as a characteristic length scale. In SA-DES, that

distance is replaced by a term proportional to the local grid spacing. This allows for a very simple implementation and does not necessarily imply a negative impact on the validity of the solution. In contrast, the method of Sanchez-Rocha has a separate LES model that is linearly blended with a RANS model. A key disadvantage of DES methods is that great care must be taken, either during grid generation or in the computation of the length scale, to avoid grid-induced separation, in which the model transitions to LES too early, in boundary layer regions of the mesh that are too coarse to support it. The Delayed DES (DDES) model successfully addresses this shortcoming by supplying a limiter that depends on the local eddy viscosity [99].

Recently, Vatsa and Lockhard investigated the application of hybrid turbulence models to aeroacoustic applications [112]. The models included a DDES model that was slightly modified to avoid spurious upstream eddy viscosity and the GT-HRLES model, which is developed later in this work. In general, both hybrid models were found to accurately predict mean surface pressures and surface pressure power spectral densities. Additionally, results with the modified DDES model in structured and unstructured frameworks were compared and found to be essentially identical. However, this did not indicate one way or the other whether unstructured methods pose any special challenges for hybrid turbulence models since in both cases a structured grid was used, though in the case of the unstructured method, it was treated as an unstructured mesh of hexahedral elements. A better test of equivalence would use a mesh of tetrahedra and prisms, which is more representative of a typical unstructured grid, that is carefully constructed to resemble the structured grid in terms of node and cell distribution.

2.3 Full Blade and Rotor CFD

If the modeling approximation offered by actuator blade methods is unacceptable for a particular engineering application, the individual blades can be modeled, though at greater computational expense.

In 1999, Duque et al. [29] employed the overset CFD solver Overflow to compute the flow over the NREL Combined Experiment Phase II rotor. These early Navier-Stokes

results were all at zero yaw, but included the nacelle and tower. While the results were promising in predicting rotor/tower interaction, they were hampered by poor overlap in the overset mesh. In 2003, Duque et al. [28] performed Navier-Stokes simulations on the NREL Phase VI rotor and compared that CFD analysis with the rotorcraft comprehensive code CAMRAD, which used various stall delay and dynamic stall models. The Navier-Stokes computations were shown to more accurately predict stalled rotor performance and also revealed significant spanwise flow.

Due to the complexity and expense of a full Navier-Stokes solution to wind turbine problems, various approximations have been used in CFD analyses. One approach is the hybrid method of Xu and Sankar [115], in which the Navier-Stokes equations are solved only in a small zone approximately 2 chords wide surrounding each blade. The rest of the flowfield is treated with a significantly less expensive full potential method. A free wake method is also used in the potential zone to model the wake formed by the tip vortices. This method showed promise at low wind speeds when flow separation is minimal, but at higher wind speeds, full Navier-Stokes is required.

Instead of searching for reduced-order numerical methods, other researchers have chosen to employ complete Navier-Stokes methods, but with drastically simplified models that eliminate much of the geometry except a single blade. In 2004, Le Pape and Lecanu used ONERA's code *elsA*, a structured multi-block solver, to model the NREL Unsteady Aerodynamics Experiment in an upwind, zero-yaw configuration [43]. Rather than model the entire rotor, they modeled a single blade, and as was common at the time, omitted the tower and nacelle. Furthermore, they solved the Navier-Stokes equations in a rotating frame of reference. Torque and thrust predictions in that study were in good agreement with the experimental data at lower wind speeds, before the onset of stall. But at moderate speeds, stall was predicted too early, leading to under-prediction of torque.

Though CFD has largely been perceived as too expensive for routine use on full turbines, it has seen a great deal of use in analysis of individual turbine parts, such as advanced airfoils. Stone et al. [103] used OVERFLOW to evaluate the aerodynamic performance of flatback airfoils, a new type of airfoil with a very blunt trailing edge designed to better

carry the structural loads of a large turbine. They compared the flatback performance, as predicted by a standard Spalart-Allmaras turbulence model, a Detached Eddy Simulation model, and a hybrid RANS-LES model, to that of the conventional airfoil. As expected, the DES and HRLES models were able to capture the inherently 3-D physics of the airfoil wake, unlike the S-A model. Given the great influence of 3-D and rotational effects on the aerodynamics of even conventional rotors [83], this study highlights the importance of accurate turbulence modeling for physically realistic simulations.

Chao and van Dam [22] used OVERFLOW to study the torque produced by a notional flatback version of the NREL Phase VI rotor compared to the actual rotor. They found that flatback airfoils are viable alternatives to conventional airfoils in large rotors in that the torque is not seriously degraded compared to a conventional rotor at low to moderate wind speeds. They also noted that, due to the low freestream velocities, low-Mach preconditioning was essential to achieving accurate solutions when using a compressible flow solver. However, due to limitations in the LMP method in OVERFLOW at the time, they relied on a steady-state rotational source term formulation of the Navier-Stokes equations, rather than moving the rotor blades through a background mesh. While they obtained useful results, this method cannot be applied to yawed rotors, since the flow over a yawed rotor is inherently unsteady, even when fully attached.

Using an incompressible overset method, Zahle et al. [117] showed that CFD can capture the unsteady physics associated with interactions between tower and rotor wakes. In that study, a two-bladed NREL Phase VI rotor was set downwind of a simplified tower, without a nacelle. The rotor lacked a physical connection to the tower and instead "hovered" just upstream of it. However, the major physical phenomena of blade-wake interaction were captured since the tower obstruction was present. Further accuracy was obtained by including a simple ground plane. As is common in other work discussed earlier, agreement with experimental performance data was reduced at higher wind speeds, an effect attributed to turbulence modeling.

In a second study by Zahle et al. [118], the same incompressible overset method was applied to an upwind rotor with inflow shear to emulate an atmospheric boundary layer.

They found that the inflow shear resulted in a lag in the blade loads compared to a case with uniform inflow as the blades pass in front of the tower. Furthermore, the blades exhibited dynamic stall effects in that region. Finally, they found that with the shear inflow, the wake dissipates faster downstream of the rotor due to mixing between the upper and lower halves of the wake.

One of the earliest unstructured CFD analyses of the wind turbine rotor was conducted by Sezer-Udol and Long [87]. Their simulations of the NREL Phase VI rotor were strictly inviscid, and blade motion was enabled by rotating the entire mesh, which prevented the inclusion of a tower. Though their inviscid method did capture some of the massive separation present at 15 m/s, rotor thrust was in all cases nearly double the measured value.

Finally, Potsdam and Mavriplis [74] modeled the NREL Phase VI rotor using the unstructured Navier-Stokes code NSU3D. They modeled the rotor in a zero-yaw configuration using both steady-state and time-accurate formulations, and a range of different surface mesh topologies. They also investigated adaptive mesh refinement to better capture the turbine wake. They found that performance results obtained with the unstructured code compare favorably with those obtained with the structured overset solver OVERFLOW, but that due to the problem of high aspect ratio boundary layer cells, a problem shared by all unstructured codes, unstructured grids may require more nodes for the same level of accuracy. Using the Spalart-Allmaras turbulence model, the onset of stall was delayed and the power over-predicted in the early stall regime, indicating the need for more advanced turbulence models.

2.4 CFD-CSD Coupling

Coupling between computational fluid dynamics (CFD) and computational structural dynamics (CSD) tools has been used extensively in the rotorcraft community to predict unsteady blade loads with greater accuracy than traditional comprehensive codes [77]. Sophisticated structural models are often used in blade element-momentum (BEM) codes for wind turbine design. Furthermore, Navier-Stokes CFD has become more common in recent years for analysis of particular design points. However, the two are rarely used in combination.

Since there has been little use of CFD/CSD coupling methods in wind turbine applications, this section largely focuses on the development of these methods for rotorcraft. Fortunately, the coupling methods originally developed for rotorcraft analysis can be adapted for wind turbines. However, there are some important differences in assumptions that need to be addressed before applying rotorcraft coupling methodologies to wind turbines. These can include non-uniform freestream velocity due to wind shear and turbulence; flexible tower effects, resulting in an oscillating hub position; and construction defects, which lead to rotors with slight variations in structural and aerodynamic properties from one blade to another.

2.4.1 Definitions

Coupling refers to the simultaneous (or nearly simultaneous) solution of the equations describing the flow over the rotor and the deformation of the blades. If the two sets of equations are not solved simultaneously, then the CFD and CSD tools that handle each set must communicate in some manner. The CSD code computes time-accurate blade deformations and provides those surface motions to the aerodynamic code, which uses them to modify its boundary conditions. The CFD code computes the flow over the entire flowfield and integrates those loads on the blade surfaces. It provides those loads to the CSD code, which uses them as forcing functions for the structural model. There are two principle types of coupling:

Loose coupling means that the fluid and structural equations are solved using separate tools that communicate at some interval greater than the time step of one of the codes. Common schemes have the two codes exchanging loads and motion once per rotor revolution or once per blade passage (n_b times per revolution, where n_b is the number of blades). The two codes might not, and often do not, use the same time step.

Tight coupling means that the two codes are still executed separately but they communicate once per time step. Clearly, this can become computationally expensive, depending on the code communication methods used (file I/O vs. library calls). Due to the computational expense, loose coupling is usually favored in situations where the airloads and structural deflections are expected to be periodic. Tight coupling is required in other situations, such

as in maneuvering flight.

A third possible type of coupling is *intimate coupling* or *full coupling*, in which the fluid and structural equations are solved simultaneously, in the same solver [8]. Though such an approach has the advantage of having no costs associated with communication of loads and motion, this type of coupling is almost never used in practice for rotating systems. Intimate coupling would require creating a dedicated analysis tool that would be extremely complex.

2.4.2 Development of Early Coupling Methods

Due to the computational expense involved, the CFD half of the coupling problem has always been the limiting factor. For this reason, the development of coupled CFD/CSD methods has essentially tracked the development of CFD for routine aerospace analysis tasks. First full potential methods were applied to isolated blades, then on full rotors. Then the Euler equations were solved on isolated blades, then on rotors. More recently, full Navier-Stokes simulations have become the norm.

In the early 1980's, Johnson developed the comprehensive analysis tool CAMRAD. For computational efficiency, it used lifting line aerodynamics and a prescribed wake model. In 1984, Tung et al. performed limited coupling between CAMRAD and the full potential code FPR in order to capture transonic effects on the advancing side of the rotor that are beyond the scope of a lifting line model [110]. Because these transonic effects only exist near the tip, and because the full potential computations were substantially more expensive than other parts of the analysis, the full potential "zone" was limited to the tip region. In 1989, Bridgeman et al. [18] improved on the CAMRAD/FPR coupling by introducing an integral boundary layer model to FPR in order to more accurately determine viscous drag on the advancing side.

A common feature of early coupling methods was that rather than deforming the blade grids and thus mimicking the physical deformation, they relied instead on modifying the boundary conditions used in the CFD solver. Typically, blade motions would be used to set up a transpiration boundary condition in which flow was allowed normal to the unmoving blade surfaces [18]. In addition, they avoided using CFD throughout the computational

domain, instead focusing on selected areas that were difficult to handle with lifting line aerodynamics. Due to the limited coverage of the CFD zone, the rotor wake was excluded, and its influence was accounted for using a varying inflow angle.

In 1994, Smith developed an aeroelastic prediction method coupling a Navier-Stokes solver with a structural solver for hovering rotors [92]. In this work, a single blade was structurally modeled as steady-state beam problem. The flowfield was computed using a fourth order spatial algorithm with a compact stencil, which allowed for solutions using 40-50% fewer grid nodes than would have been required in a conventional second order algorithm, which lead to corresponding reductions in CPU time.

In the early 1990s, overset CFD methods gained popularity due to the ease with which they could model complex real-world geometries. One of the first widely used overset Navier-Stokes solvers was NASA's OVERFLOW. In its original form, it was capable only of static overset simulations. In 1994, OVERFLOW-D was released with the capability to handle arbitrary grid motion. This innovation made it very popular for rotorcraft CFD. Indeed, while rotorcraft CFD has by no means been restricted to any solver in particular, a good overview of its development can be had by focusing on OVERFLOW simulations alone.

Shortly after OVERFLOW-D was developed, Bauchau and Ahmad [8] performed some of the first examples of tight CFD/CSD coupling between OVERFLOW-D and the flexible multi-body dynamics code DYMORE. Unlike other comprehensive analysis tools in existence at that time (like CAMRAD) which operate in the frequency domain and rely on a modal analysis of the structural model, DYMORE works strictly in the time domain and integrates the discretized equations of motion directly. Operating in the time domain is a requirement for tight coupling since all CFD solvers also operate in the time domain. The work of Bauchau and Ahmad is also distinguished from other aeroelastic work discussed here so far in that they solved the thin-layer Navier-Stokes equations on a computational domain that covered the entire rotor without reliance on a wake model. However, due to the large computational expense, no rotor trim was performed; control angles from experimental data were used instead. Even without trimming, their computed airloads showed clear differences from rigid-blade computations, indicating the importance of aeroelastic effects.

In 1997, another tight coupling analysis, that of Lee, Saberi, and Ormiston [44], highlighted some of the algorithmic issues that would be heavily investigated in coming years. They tightly coupled the comprehensive analysis tool 2GCHAS with the full potential solver FPX. As in previous studies that applied a full potential solver rather than Euler or Navier-Stokes CFD, they had only partial CFD coverage of the computational domain. Previous loose coupling results had difficulty achieving convergence of the coupling process, and as stated by Lee et al., "it was suggested that the loose coupling procedure was inadequate when pitch moments/torsional dynamics were included." Overcoming this difficulty was the reason tight coupling was used for their analysis. Even so, they still experienced torsional instabilities, proving that the coupling algorithm was not at fault. Instead, they found that errors in the pressure distribution at the trailing edge of the blades contributed to negative pitch damping.

2.4.3 Current State of the Art

In a typical coupling problem, the rotor collective and cyclic pitch controls are not known *a priori*. In a procedure called *trimming*, the controls must be adjusted until the mean integrated hub loads match certain target values. Typically, the collective and cyclic pitch are adjusted to match thrust and rolling and pitching moments. Trimming creates particular difficulties for the stability of a loose coupling procedure. Since coupling is performed at some interval greater than the CFD time step, and the CFD loads are held constant throughout a coupling iteration, the loads applied to the structural model are not able to change in response. Therefore, there is no aerodynamic damping in the structural equations. In the last few years, a particular loose coupling algorithm that addresses these problems of poor trim and convergence has become the norm in CFD analysis of rotorcraft. In this so-called *delta trimming* method, the loads applied to the structural model are actually the sum of loads from a low-fidelity lifting line model and a *delta load* equal to the difference between the lifting line and CFD loads [77, 26]. At coupling iteration k, the applied forces

and moments on the structural model, F^k, are given by:

$$F^k = F^k_{LL} \quad + \Delta F^{k-1} \tag{1}$$

$$= F^k_{LL} \quad + \left(F^{k-1}_{CFD} - F^{k-1}_{LL}\right) \tag{2}$$

$$= F^{k-1}_{CFD} + \left(F^k_{LL} - F^{k-1}_{LL}\right) \tag{3}$$

F_{LL} and F_{CFD} are forces obtained from the lifting line and CFD models, respectively. As the solution progresses, F_{LL} should stop changing, so at convergence the applied load would simply be $F^k = F^{k-1}_{CFD}$. Since the lifting line loads are free to change at any time step, they can provide structural damping.

One of the first studies to use the delta trimming method was that of Potsdam et al. [77], in which OVERFLOW-D was coupled with CAMRAD-II and RCAS and applied to the UH-60A rotor. CFD was used throughout the computational domain, and the blade grids were deformed elastically, mimicking the physical deformation that occurs in flight. This is in contrast to earlier studies that relied on transpiration boundary conditions on the blades to include the effects of structural deformation. Motion and loads communication between the codes was accomplished via the exchange of specially formatted files on disk (which has since become the norm for loose coupling). These computational results were compared against experimental airloads from the extensive UH-60 flight test database. Three forward flight conditions were analyzed: the high speed case known as c8534, the low speed case c8513, and the high thrust case c9017. For all cases, convergence of the trimming algorithm was fairly rapid.

The high speed case is less demanding from a CFD perspective because the wake, which can be difficult to capture without very fine grids or high order algorithms, is convected away from the rotor quickly enough that it has little effect on airloads. Good agreement with experimental sectional normal force C_nM^2 was found, indicating the success of the full Navier-Stokes approach. While the overall trend of the sectional pitching moments C_mM^2 was good, their magnitude was consistently over-predicted inboard and under-predicted outboard. In addition, there was a 25^o phase shift in the pitching moments.

The low speed case c8513 is dominated by blade-vortex interactions (BVI), which are

notoriously difficult to capture with CFD since most schemes rapidly diffuse the vortices. Nevertheless, both the magnitude and phase of the normal force and pitching moments were well-predicted, particularly the vibratory components with a natural frequency of 3/rev and higher.

The high thrust case c9017 is at the edge of the UH-60 flight envelope and is dominated by dynamic stall events. Phase shift effects were quite evident in this test case. Though Potsdam et al. had no difficulty with convergence in this case, others have. Large changes in control angles may not produce large changes in rotor loads since the blades may simply stall with the extra angle of attack.

Datta et al. [26] also applied the delta loads method to the UH-60 database by coupling TURNS and UMARC. For reasons of computational expense, the CFD method was somewhat simpler than that of Potsdam et al. TURNS only models a single blade; the effect of the wake from other blades was included through a Bagai-Leishman free wake model. They chose to focus on the prediction of vibratory airloads, which are particularly important for both fatigue analysis of the rotor and airframe and passenger comfort (in civilian aircraft).

Datta et al. [26] experienced problems with both the magnitude and phase of airloads similar to those experienced by Potsdam et al. In discussing poor airload predictions on the advancing side in the high speed case c8534, they explained that poor pitching moment predictions are largely to blame. Pitching moment is the major driver in vibratory loads due to its influence on the elastic torsion and hence the sectional angle of attack. Pitching moment is much more sensitive to inaccuracies in the flow solution since it is computed in large part by multiplying surface pressure forces by some moment arm, which can have a magnifying effect. Despite these deficiencies, they found that the use of CFD yielded a great deal of improvement over lifting line and free wake aerodynamics because CFD is able to handle the transient shocks that occur on the advancing side in a way that other tools simply cannot.

More recently, the problem of BVI has been tackled further by Lim et al. [46] using loose coupling between OVERFLOW-2 and CAMRAD-II applied to the HART-I and HART-II rotors and by Makinen et al. [50] using OVERFLOW-2 and RCAS applied to HART-I. Again,

good convergence of the trim algorithm was observed. In both studies, results concerning grid quality were presented. In the former, it was found that a "standard" grid of 20 million points showed great improvement in airloads prediction over a "coarse" grid with only about 5 million points. However, a "fine" grid with over 100 million points showed very little improvement over the the standard grid.

The studies discussed thus far all used algebraic or one equation turbulence models such as Baldwin-Barth and Spalart-Allmaras. The work by Makinen et al. [50] went further in using a newly developed HRLES turbulence model [88], which is a more physics-based approach to turbulence modeling. With HRLES and higher-order spatial discretization, they were better able to take advantage of very fine off-body grids and capture more of the vortical wake. Of course, such results come at an extreme computational cost.

All of the CFD/CSD coupling efforts discussed so far have used structured CFD methods. In principal, the CFD mesh type should have no bearing on the applicability of a particular coupling methodology. In practice, unstructured grids are more difficult for coupling because very few assumptions can be made about the mesh geometry and so more rigorous procedures must be used for common tasks such as integrating airloads and deforming meshes. For example, when determining the loads to apply to the CSD model, pressure and viscous forces are usually calculated on spanwise slices of the surface grid. In a structured grid, slices are usually parallel to constant i, j, or k lines, making it straightforward to determine which surface faces contribute to the load on an airstation. With an unstructured grid, a detailed search procedure is needed to identify such faces, either through an exhaustive search of every boundary face or the use of some auxiliary data structure facilitating spatial search.

Abras developed a number of important techniques for using the delta coupling method of Potsdam [77] in an unstructured grid solver. Specifically, she coupled the unstructured solver FUN3D with the flexible multibody dynamics code DYMORE to predict UH-60 forward flight cases [4, 3]. These simulations illustrate the extra care that must be taken in an unstructured method. For example, in a structured solver the volume grid might be deformed by simply translating and rotating each spanwise grid plane, but this is not feasible

in an unstructured solver. Abras used a "spring analogy" to determine how the grid should be deformed in response to the lifting line deflections provided by DYMORE. She found that to the deformed mesh from being excessively stretched, it was necessary to apply those deflections as a combination of a small elastic deformation and a larger rigid motion.

Biedron and Lee-Rausch investigated loose coupling with FUN3D and CAMRAD-II, applied to the HART-II rotor [16]. In addition to demonstrating another variant of the loose delta coupling technique, they performed a detailed study of the grid resolution and temporal accuracy required to achieve acceptable correlations with experimental data. In particular, they found that resolution of the prominent blade-vortex interaction (BVI) events that occur in the HART-II descent cases require much finer grid resolution than that used by other researchers in forward flight cases.

In nearly all coupled rotorcraft simulations, it is assumed that while the blades may bend and twist, their cross-sections remain fixed. In 2009, Thepvongs et al. [108] developed a coupling algorithm intended for studying rotors with morphing airfoils. Instead of a beam representation, the blades were modeled using a 3-D finite element model capable of large deformations. The structural code included a finite-state aerodynamic model, and coupled aeroelastic predictions compared favorably with predictions using only the lower order aerodynamics.

CHAPTER III

DESCRIPTION OF THE CODES

This chapter contains descriptions of the basic methods used by the two CFD solvers used in this work, OVERFLOW and FUN3D, as well as the flexible multibody dynamics tool DYMORE.

3.1 OVERFLOW

OVERFLOW [19] is an overset structured grid CFD solver developed primarily by NASA. The use of overset grids simplifies modeling of flows around complex geometries or moving bodies since a given grid need only cover part of the geometry rather than the entire flowfield. OVERFLOW solves the unsteady, compressible Navier-Stokes equations using an implicit finite-difference formulation. By default, fluxes are computed using a 2nd order accurate central difference scheme, though spatial accuracy as high as 4th order is available. A Roe upwind scheme is also available for the inviscid fluxes. Unsteady solutions can be 1st or 2nd order accurate in time. To extend the range of Mach numbers over which the compressible method can be used, unsteady low-Mach preconditioning (ULMP) [76] is available. OVERFLOW has been used successfully for fixed-wing, rotorcraft [5, 53, 47], and wind turbine applications [29, 28]. A number of turbulence models are available in OVERFLOW, including k-ω SST and GT-HRLES, which was implemented as part of the DARPA Helicopter Quieting Program [30].

3.2 FUN3D

FUN3D is NASA Langley's code for solving the Navier-Stokes equations on tetrahedral or mixed element unstructured grids [17, 6]. It can solve both compressible and incompressible flows, the latter computed using using Chorin's method of artificial compressibility [23]. It uses an implicit node-centered finite-volume formulation. Steady flows are solved using a 1st order backward Euler scheme, while time accurate schemes are solved using a 2nd order

backward differentiation formula (BDF). By default, the inviscid fluxes are evaluated using Roe upwinding. The viscous fluxes are always evaluated using a scheme that is equivalent to a central difference formulation. The resulting linear system is solved using a point-implicit red-black Gauss-Seidel scheme. A number of common turbulence models are available, including k-ω SST and DES.

Overset grids can be used in FUN3D in conjunction with DiRTlib (Donor Receptor Transaction Library) [62] and SUGGAR (Structured, Unstructured, and Generalized Grid AssembleR) [63]. The primary use of overset methods in FUN3D is moving body problems [15] since unstructured grids already provide a powerful tool for handling geometric complexity. Similar to OVERFLOW, FUN3D has been used for a very wide range of applications, including rotorcraft [67, 78].

3.3 DYMORE

DYMORE is a multi-body finite element code developed by Bauchau and others [10]. DYMORE contains a library of primitive elements, such as beams, springs, hydraulic dampers, and various types of joints, for which the equations of motion can be stated exactly. A model consists of one or more primitive elements connected so that the connections enforce the boundary conditions for each element. All equations of motion are expressed in a single inertial frame and are integrated in the time-domain, in contrast to other codes, which often express the equations in body-fixed frames and integrate in the frequency domain. The formulation used in DYMORE is more general, which makes it applicable across a broader range of applications, rather than being restricted to a few cases (e.g. loose coupling). The code is well validated and in active use in the rotorcraft industry.

CHAPTER IV

IMPROVED ACTUATOR BLADE METHODS

O'Brien demonstrated that an actuator blade methodology can provide a reasonable approximation of the unsteady loads on a helicopter fuselage due to the rotor wake [67, 69]. As discussed earlier, these methods model the influence of the rotor on the flowfield by adding source terms to the Navier-Stokes equations that act as body forces.

A flowchart representation of the actuator blade algorithm is given in Figure 3. First, the new positions of the actuator sources, which rotate through the mesh in clouds that reproduce the approximate blade planform as shown in Figure 4, are computed. Then a search is done to find the nearest neighboring mesh vertex to each source. Since each processor stores *all* of the actuator sources, but only a subset of the mesh, a given source may lie outside the mesh partition. In that case, the source's true nearest neighbor actually lies on another partition. To deal with that situation, the processors exchange the distance they computed from each source to its nearest vertex. If another processor has a smaller distance, that source is marked as belonging to another partition, and its influence is not included on that processor. This "multiple partition check" is handled with an efficient `MPI_Reduce` operation. Following that check, the state variables at the vertex associated with each source are used to computed the local effective angle of attack. That angle of attack is then used in a blade element momentum (BEM) model to compute source strengths. Finally those source strengths are added to the residual and Jacobian, and the linear system is solved. A more complete discussion of this overall process can be found in Ref. [67].

Source-to-vertex association entails a nearest neighbor search. Originally, a simple exhaustive search was used in which the distance between each source and each mesh vertex is computed, making it an operation with $O(N_v N_s)$ complexity, where N_s is the number of actuator sources, and N_v is the number of mesh vertices resident on a particular processor.

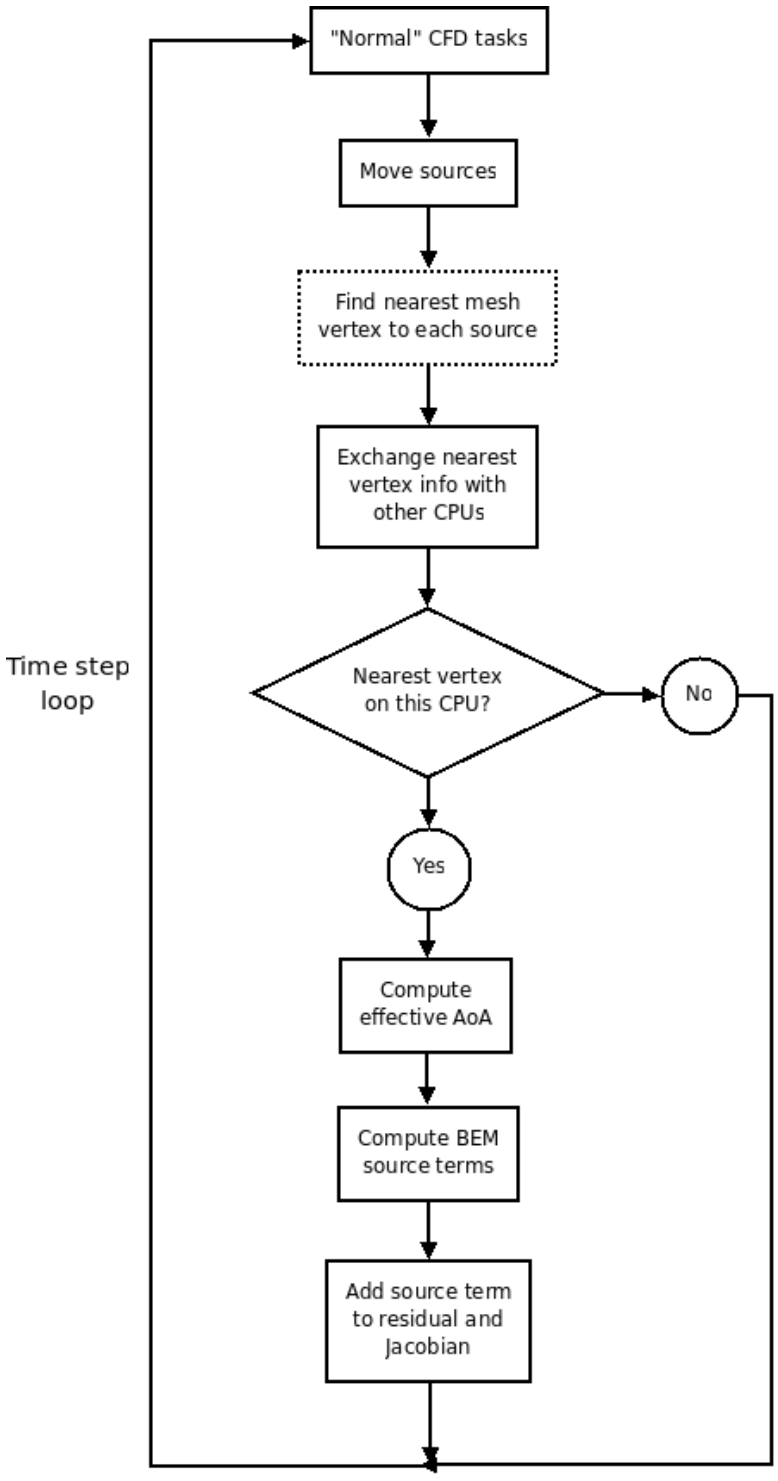
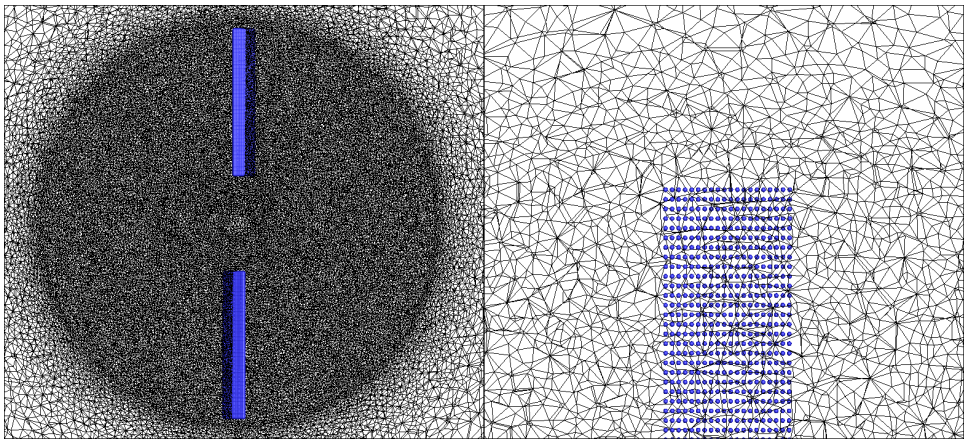


Figure 3: Flowchart of the actuator blade solution algorithm. The box with a dashed border, indicating the search portion, is the only part that was altered.



(a) Complete rotor (b) Close-up of tip

Figure 4: Actuator sources (blue) embedded in mesh. Each source must be associated with the nearest mesh vertex.

In a typical case, N_v might range from 30,000 to 200,000, and N_s might be on the order of 2,000 per blade. Thus, for some configurations, this search can become a very expensive operation. Indeed, O'Brien found that the cost of an actuator blade solution using this implementation can approach the cost of a solution with overset blades [69].

4.1 Kd-tree search algorithms

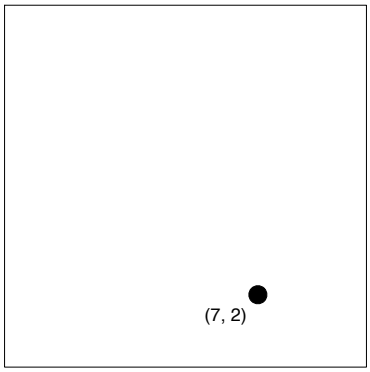
Nearest neighbor searches have been thoroughly studied in the computer science literature, and many search algorithms exist that are faster than the exhaustive search described above, at least in three dimensions. *Space-partitioning* methods are a class of algorithms in which the set of points being searched is recursively divided into smaller subsets. The divided set may be represented by a tree-based data structure like an *octree* or a *kd-tree*. Though kd-tree literally means "k-dimensional tree," it usually refers to a specific type of binary tree in which each level of the tree is split along one Cartesian direction, with the direction alternating in an x,y,z fashion through the tree. Such trees are known to be among the best data structures for nearest neighbor searches in two or three dimensions [13]. In general, a kd-tree nearest neighbor query has $O(\log N)$ complexity, where N is the number of nodes in the tree [32].

4.1.1 Tree construction and nearest neighbor queries

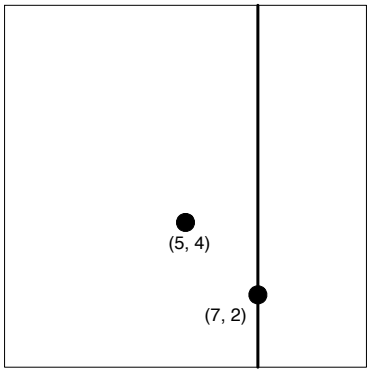
For purposes of actuator source-to-vertex association, a node of a kd-tree contains a list of at most m_v mesh vertices, including the vertices' array indices and their positions in space, as well as pointers to left and right child nodes. Each tree node also has a split direction, d, and a split point, $\vec{s}$. All vertices in the left child have $p_d \leq s_d$, and all vertices in the right child have $p_d > s_d$, where $\vec{p}$ is a vertex's position in space.

Kd-trees are constructed recursively, so the construction procedure works identically at any depth in the tree. Since the mesh vertices are static, the tree is constructed just once, at the beginning of the simulation. The process begins with a node being passed a list of n_v mesh vertices (at the beginning, all vertices are passed to the root node). If $n_v \leq m_v$, all of the vertices are inserted into the node's list, and control returns to the parent node. If $n_v > m_v$, the vertices are sorted by their position in the split direction, d. Once sorted, the lower half of the vertices are sent to the left child, and the upper half are sent to the right child. The median point alone is inserted into the current node. The split direction alternates through the tree, so that if a node is split in direction, d, its two children are split in the `mod(d,3)+1` direction (assuming 1-based indexing).

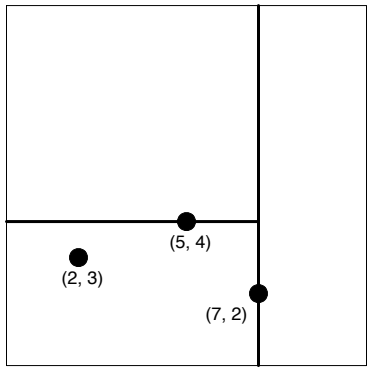
For example, consider the following list of six 2-D vertices: $(4,7)$, $(9,6)$, $(5,4)$, $(2,3)$, $(7,2)$, $(8,1)$. Assume $m_v = 1$, meaning that only one vertex may be stored in each tree node, and that the initial split direction is x. First, the list is sorted in the x-direction. The median of the list is $(7,2)$ and is inserted into the root node of the tree (Figure 5(a)). That vertex becomes the split point for the root node. The nodes to the left of $(7,2)$ are now sorted in the y-direction. The median of that sublist is $(5,4)$, which is inserted into the left child of the root (Figure 5(b)). The only vertex to the left of (below) $(5,4)$ is $(2,3)$, so it goes into the left child of the root's left child (Figure 5(c)). Vertex $(4,7)$ is handled similarly (Figure 5(d)). Then control moves to the right half of the original sublist: $(8,1)$ and $(9,6)$. That sublist is sorted in the y-direction. The median is $(9,6)$, which is inserted into the right child of the root, and then that child is split at that point (Figure 5(e)). The only remaining node is inserted into the left (bottom) child of the root's right child (Figure 5(f)).



(a) Insertion of the first vertex

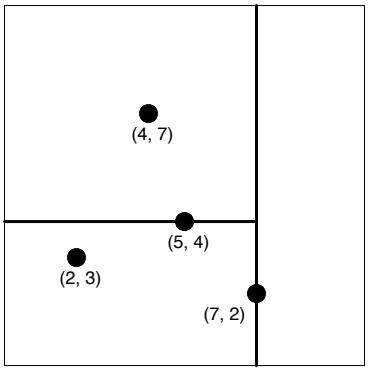


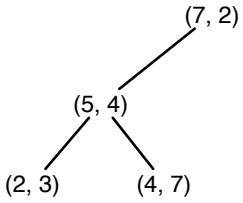
(b) Root node split in x-direction; 2nd vertex inserted into left child.



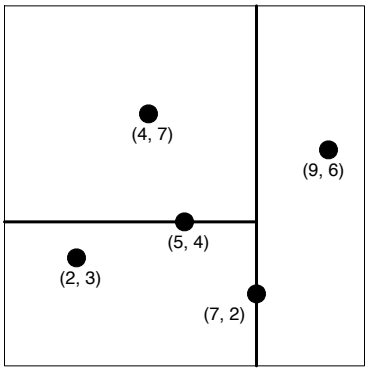
(c) Left child of root split in y-direction; 3rd vertex inserted into "left" child of root's left child.

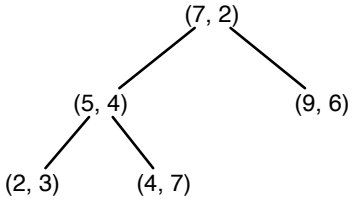
Figure 5: Graphical depiction of the algorithm for building an $m_v = 1$ kd-tree. The partioning of physical space is shown on the left, and the data structure is shown on the right.



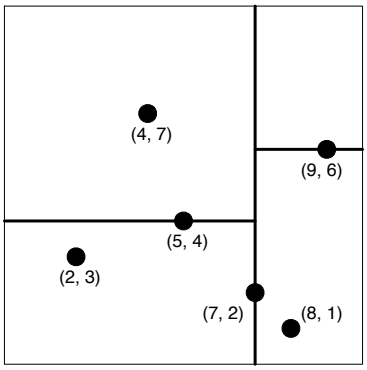


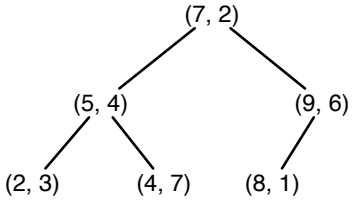
(d) 4th vertex inserted into "right" child of root's left child.





(e) 5th vertex inserted into right child of root.





(f) Root's right child has been split in y-direction; 6th vertex inserted into its "left" child.

Figure 5: Graphical depiction of the algorithm for building an $m_v = 1$ kd-tree (continued). The partitioning of physical space is shown on the left, and the data structure is shown on the right.

Searching a kd-tree for the vertex nearest to a query point, $\vec{q}$, can begin at any node, though typically it begins at the root node. The number of points stored in a node and its children is n_v. If the node has $n_v \leq m_v$, it is a terminal node (having no children), and a simple exhaustive search is done over its vertices. That is, the distance squared between q and p is computed for each vertex and then compared to the smallest distance found so far. If it is smaller, it becomes the new "best" distance, and that vertex's index is saved. Control then returns to the parent node since terminal nodes have no children.

If the node has $n_v > m_v$, the distance squared between $\vec{q}$ and the one point actually stored in the node is computed and compared to the minimum distance encountered so far. If the query point lies to the left of the node's split point ($q_d < s_d$), the left child is recursively searched. Otherwise, the right child is searched. After that search, the distance from the query point to the plane separating the two children is computed. If it is smaller than the minimum distance encountered so far, it is possible that the nearest vertex to the query point lies just across that plane in the child that was not searched previously. In that case the other child is searched recursively as well. Finally, control is returned to the parent node.

For example, consider the example tree constructed earlier and a query point $(1, 5)$. A search for the query point's nearest neighbor begins at the root node, with vertex $(7, 2)$. The distance squared from the query point to that vertex is 45 (Figure 6a). Since the query point is to the left of the root node's split point, we now search the root's left child, which contains the vertex $(5, 4)$. The distance to that vertex is 17, which is the best encountered so far (Figure 6b). Since the query point is to the "right" of (above) the split point, we now search the right child, with vertex $(4, 7)$. That distance is 13, which is better still than the previous closest vertex (Figure 6c). Since this node has no children, we now check the distance to the plane separating it from its sibling. The distance to the plane is 1, which is less the the distance to the best vertex found so far (Figure 6d). Therefore, a vertex lying just on the other side of the plane could be a nearer neighbor, and so we search the sibling. The vertex contained in that sibling has a distance of 5 from the query point, and so becomes the best candidate (Figure 6e). Now control returns to the parent node, with

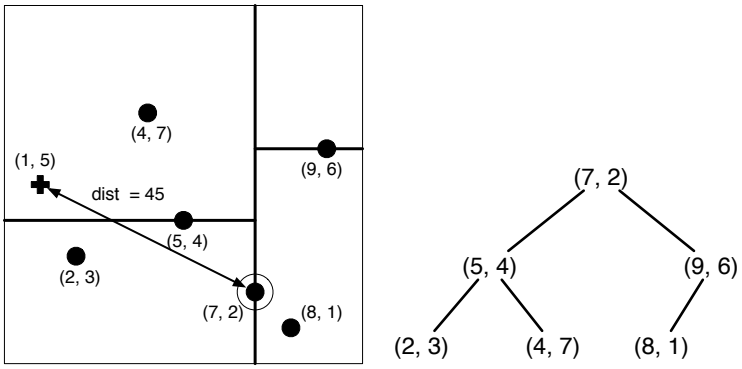
vertex $(5, 4)$. Both its children have been searched, so control returns to the root node. The distance from the query point to the root node's split plane is 36, which is greater than the distance for the best candidate, and so the right child of the root node need not be searched (Figure 6f). The nearest neighbor of the query point is $(2, 3)$. Since the search never descended into the right tree, the expensive distance function was never evaluated for those vertices.

4.1.2 Performance measurements

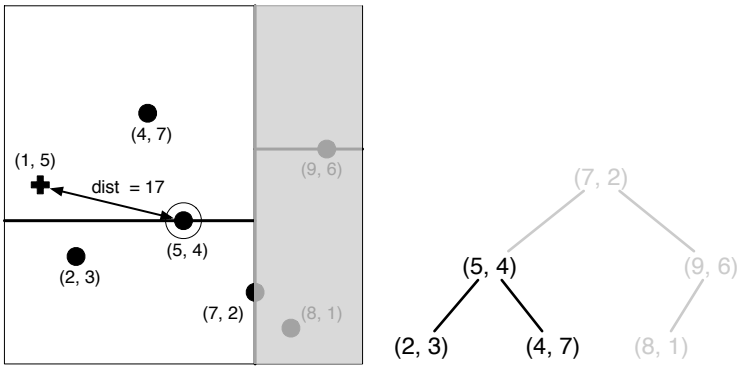
Performance of a kd-tree search will depend on several factors. In order of importance, they are the total number of vertices contained in the tree, N_v; the maximum number of vertices contained in any one node, m_v; and the efficiency with which the distance function is evaluated. The third factor is addressed by always using squared distances so as to avoid computing square roots and by computing distances inline without a separate function call.

To determine how the search algorithm scales with the maximum number of vertices allowed per tree node, a simple scaling study was done on a mesh with 154,082 vertices, varying m_v. Altering m_v essentially changes the number of nodes in the kd-tree. Since kd-tree searches scale logarithmically with the number of nodes, m_v was changed by powers of two. Figure 7 shows the ratio of the average cost of a kd-tree search to an exhaustive search vs. m_v. In this limited test, the best performance was obtained with m_v set to either 64 or 128 (the difference between the two being negligible).

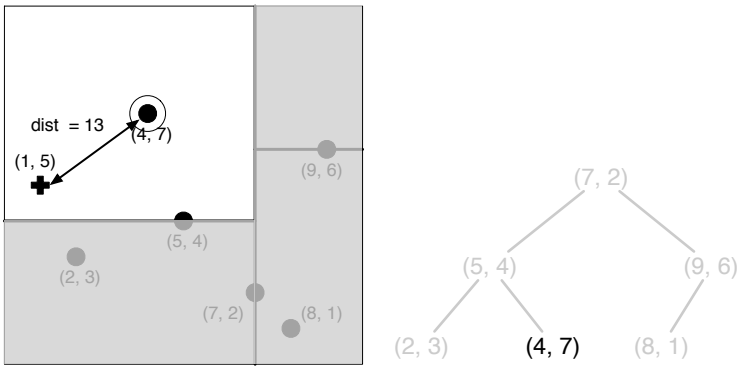
To determine the importance of N_v and m_v, several actuator blade simulations were undertaken on two similar grids. The grids contained about 2.4 and 4.5 million vertices, respectively. Both grids contained no solid boundaries. Mesh points were clustered in a region between the rotor disc and another disc about three rotor diameters downstream. The number of vertices in each CPU's tree was altered by varying the number of CPUs used in each simulation. The number of vertices per processor ranged from 17,651 to 282,420. For each of those node counts, simulations were conducted using the pre-existing exhaustive search; a kd-tree search with $m_v = 128$, previously identified as optimal; and a kd-tree search with $m_v = 16$, a reasonable but still sub-optimal value. The average search time, as well



(a) Compute distance from query point to root node's vertex.

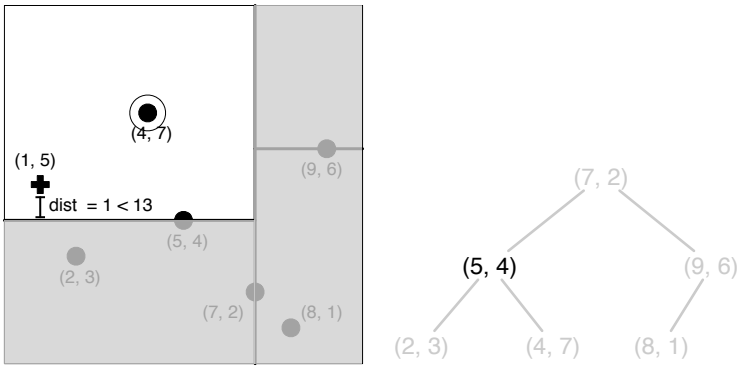


(b) Compute distance from query point to left child's vertex.

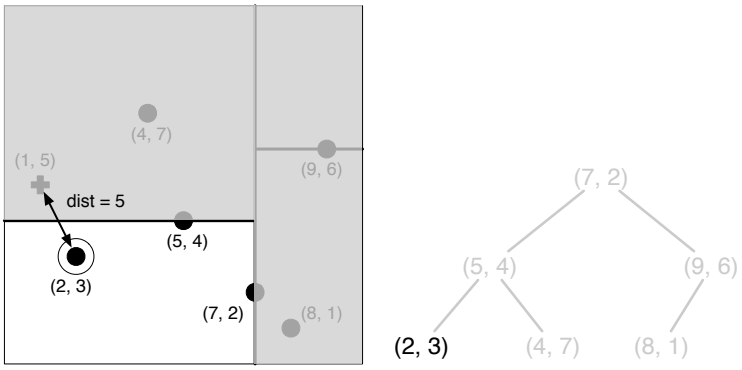


(c) Compute distance from query point to right (top) child of root's left child.

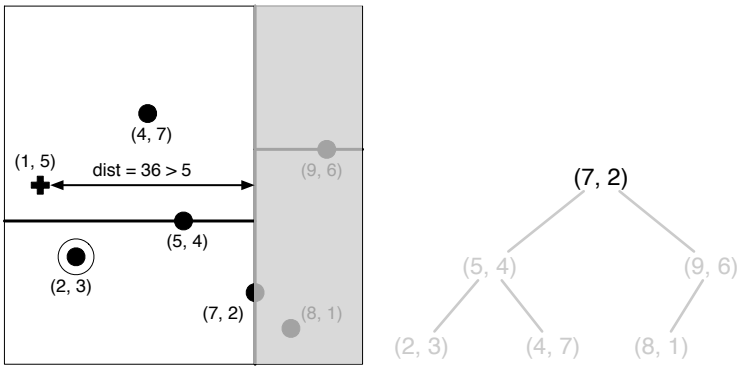
Figure 6: Graphical depiction of the algorithm for search a kd-tree wit $m_v = 1$. Vertices are represented by black dots, the query point by a black cross, and the current nearest neighbor by an unfilled circle. Gray regions are not currently being searched.



(d) Compute distance to splitting plane.



(e) Distance was less than current nearest, so search sibling.



(f) Compute distance to root splitting plane. It is greater than nearest vertex so far, so search is complete.

Figure 6: Graphical depiction of the algorithm for search a kd-tree wit $m_v = 1$ (continued). Vertices are represented by black dots, the query point by a black cross, and the current nearest neighbor by an unfilled circle. Gray regions are not currently being searched.

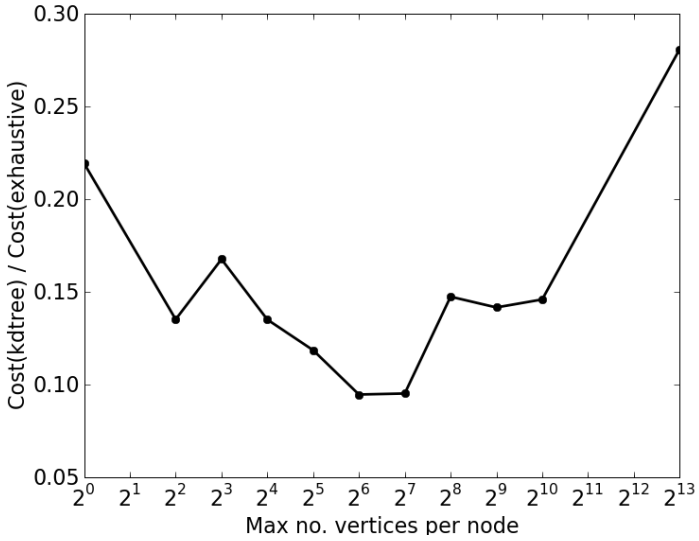
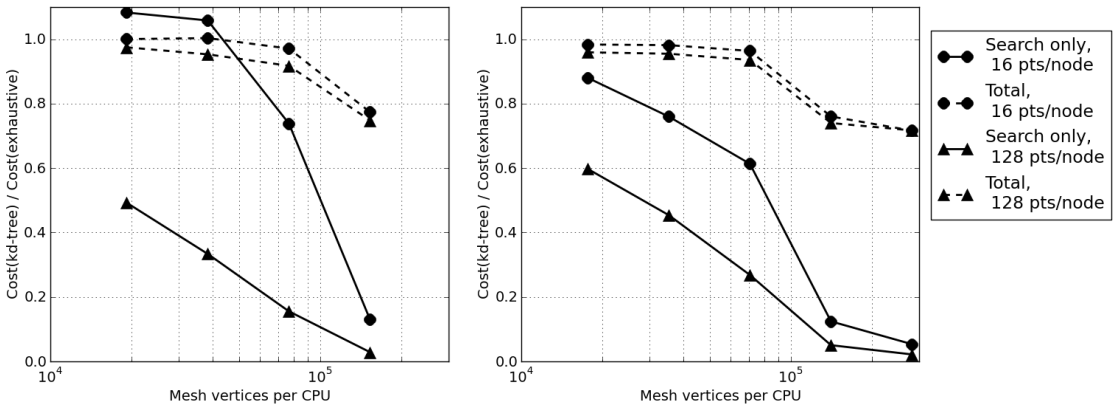


Figure 7: Ratio of the cost of a kd-tree search to the cost of an exhaustive search vs. the maximum number of vertices per CPU.

as the average total time (including activities not related to actuator blades) per time step for the kd-tree was compared to the exhaustive search.

The kd-tree to exhaustive search cost ratios are shown in Figure 8. Clearly, when the CPUs are very heavily loaded with a large number of vertices, the kd-tree is vastly superior, with less than 15% the search cost of the exhaustive search (solid lines). There are diminishing returns from the kd-tree as the number of vertices per CPU decreases (i.e., as the number of CPUs increases, since the total number of vertices is fixed). On the larger mesh (grid 2), the kd-tree is 88% as expensive as the exhaustive search. Also, at low numbers of vertices, the search procedure is overall a very small percentage of the total time per time step. This occurs because the exhaustive search makes very efficient use of cache and avoids the overhead of function calls.

The effect of m_v, the maximum number of vertices per tree node, is striking. Allowing more vertices per tree node improves performance of the kd-tree considerably, as can be seen in the distance between circles and triangles in Figure 8. Increasing m_v allows the kd-tree algorithm to gain back some of the cache efficiency of an exhaustive search. Furthermore, it reduces the maximum depth of the tree so that fewer recursive function calls are required to reach a terminal node. Indeed, as can be seen on the smaller of the two grids (Figure



(a) Grid 1, with 2.4 million vertices

(b) Grid 2, with 4.5 million vertices

Figure 8: Ratio of the cost of the kd-tree search to the cost of the exhaustive search. Solid lines indicate the cost ratio for the search portion of a time step alone. Dashed lines indicate the cost ratio for a complete time step.

8(a)), using 128 vertices per node rather than 16 ensures that the kd-tree search is always less expensive than exhaustive search.

In the Fortran implementation of the present search algorithm, a kd-tree node is defined according to Listing 4.1. Since the two arrays holding the list of vertices are statically allocated for speed, the memory usage for one kd-tree node is always:

$$\text{memory per node} = 4 + 4 + 4m_v + 8km_v + 8 + 8 = 24 + (4 + 8k)m_v \tag{4}$$

In all applications considered here, $k = 3$. In a best-case scenario for memory usage, each terminal node would hold exactly m_v vertices, minimizing the total number of nodes. In a worst-case scenario, each would hold only one vertex, giving the largest possible number of nodes and essentially wasting $28(m_v - 1)$ bytes per node. So memory usage will be in the range:

$$\frac{N_v}{m_v}[24 + 28m_v] < \text{total memory} < N_v[24 + 28m_v]\,\text{bytes} \tag{5}$$

Currently, memory usage is within acceptable bounds. Should it become necessary to reduce memory usage, the algorithm could be modified to directly index the arrays holding the mesh vertices rather than copying them (into the `pos` variable in Listing 4.1. This would save $24m_v$ bytes per node.

Listing 4.1: Fortran definition of a kd-tree node

```
type kdnode
  integer :: dir   ! splitting direction
  integer :: npts  ! no. points in or below this node
  integer, dimension(m_v)   :: index
! indices of the points
  real(dp), dimension(k,m_v) :: pos
! positions of the points
  type(kdnode), pointer      :: left, right  ! children
end type
```

4.2 Wake prediction

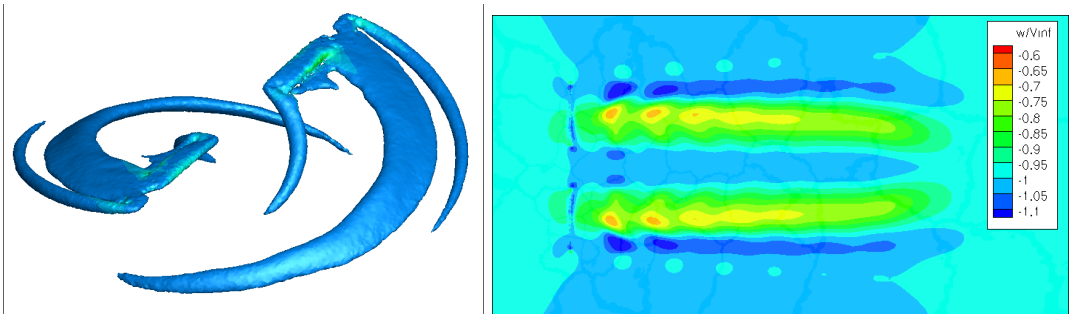
Actuator blades were used to simulate the NREL Phase VI wind turbine. A wind speed of 7 m/s was chosen since at higher speeds the flow over the blades begins to stall. Obviously, the predictive capability of an actuator blade method will suffer when the flow is dominated by viscous effects near solid boundaries or other phenomena, like stall, that depend greatly on the details of the geometry involved. Each actuator blade had 2000 sources (100 in the radial direction, and 20 along the chord, uniformly distributed) in keeping with O'Brien's recommendation [67] that the source array be finer than the grid in which it is embedded. This source distribution prevents singularities in the actuator forces and keeps tip vortices from being too diffuse. An incompressible formulation was applied as the freestream and tip Mach numbers are very low, less than 0.1. Since no solid boundaries are present, the distance function used in most turbulence models cannot be evaluated, and so the simulations were laminar. The mesh included a total of 2.4 million vertices. An overset simulation with two blades but without a tower was also conducted to compare the wake approximated by actuator sources to the wake from actual rotating blades. The background mesh of the overset simulation was identical to the mesh used in the actuator blade simulations. In both cases, a time step equivalent to 1° of azimuth per step was used.

The actuator blade method originally implemented by O'Brien uses a linear approximation of the blade twist, which is well-suited to helicopter blades. However, wind turbine blades often have very non-linear twist distributions; the NREL Phase VI blade is no exception. For that reason, the algorithm was modified to allow the user to supply the twist

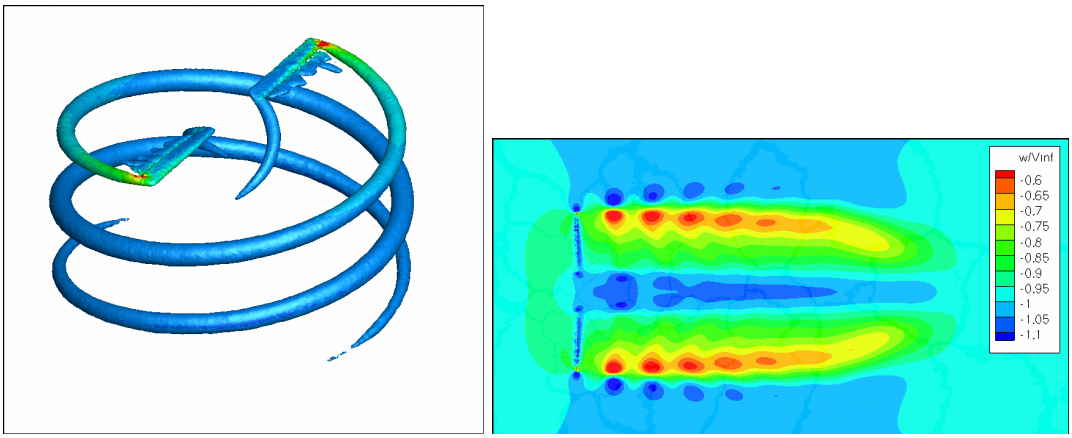
distribution in a file. The twist distribution is interpolated to get an accurate blade twist at each actuator source. Figure 9(a) illustrates the wake computed using a linear approximation of the blade twist. As the wake visualizations show, these approximations can lead to very diffuse root and tip vortices. In addition, the sign of the apparent angle of attack, which includes the blade twist, changes around 80% span, leading to tip-like effects further inboard than expected. Finally, the linear distribution results in a non-physical vortex sheet downstream of that 80% span location. Figure 9(b), which was computed using the actual twist distribution, shows none of those distortions.

Figure 9(c) shows the overset solution. Away from the blades, the overset and actuator solutions (with correct twist) are quite similar. The tip vortices predicted by the actuator blades are indistinguishable from those predicted by overset blades. The overall wake deficit is also well-captured. Naturally, the actuator blades can not mimic the starting vortex captured in the overset solution, but that is a momentary transient effect. The "jet" of higher speed flow through the center of the disk is somewhat more diffuse in the actuator solution. This can be attributed to differing geometry near the hub. The inboard edge of the actuator blades corresponds to the first radial station with an airfoil profile, 1.257 m from the axis of rotation. The actual blades have a cylindrical root attachment starting 0.508 m from the hub followed by a gradual transition to the first airfoil section at 1.257 m. This simplification was necessary because the underlying BEM model of the actuator method assumes a linear airfoil-like relationship between angle of attack and lift, which will not hold for a cylindrical section. In contrast, the overset blades match the actual NREL model blades. In addition to being closer to the hub, the cylindrical root sheds a more powerful vortex than the actuator blades.

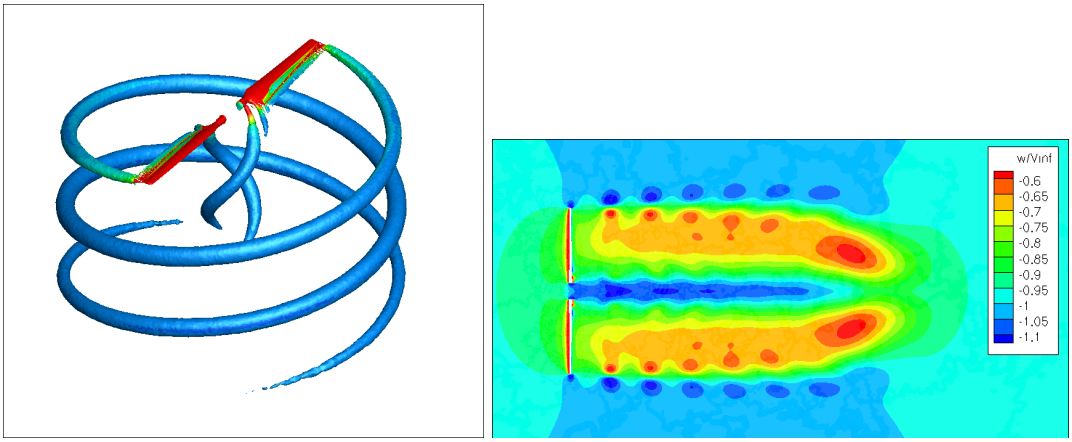
The trajectories of the vortices after leaving the blade root and tip provide another useful comparison of the actuator and overset blade methods. Figure 10 shows the axial and radial components of those trajectories. From the axial component, it is clear that the actuator and overset blade methods convect tip vortices downstream at nearly the same speed. On the other hand, the actuator blade method convects the root vortices faster. This is qualitatively evident from the contour plots of w/V_∞ in Figure 9, where the actuator blade



(a) Actuator blades with linear approximation of actual twist. Note the non-physical vortex sheet.



(b) Actuator blades with actual twist

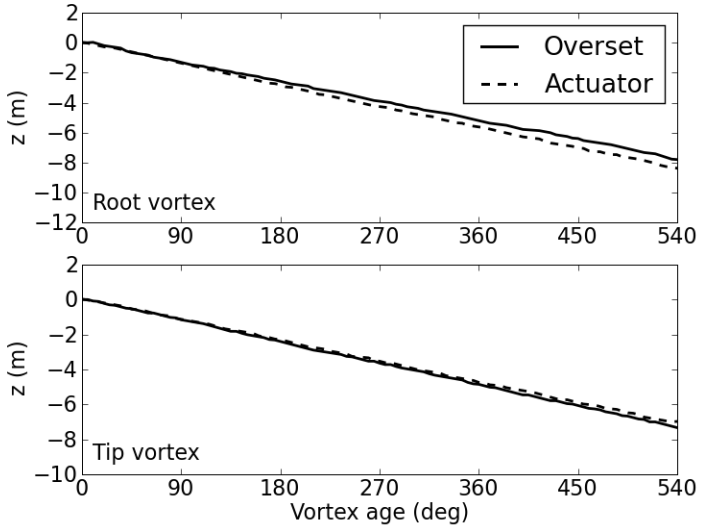


(c) Overset blades

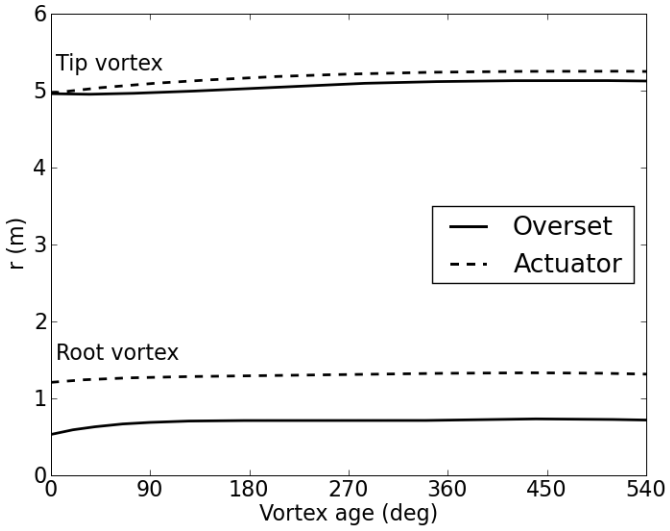
Figure 9: $Q = 0.05$ iso-surfaces and axial velocity, w/V_∞, along a centerline plane at $V_\infty = 7$ m/s, zero yaw. Q is normalized by R and U_{tip}. All cases ran four revolutions at 1° of azimuth per step.

results have a larger high speed core. The principal difference in the radial component of the root vortex trajectory is an inboard offset of about 0.6 m for the actuator blade solution. This is mostly due to the omission of the cylinder-to-airfoil transition in the actuator model, which moves the root 0.75 m inboard.

The actuator blade method was originally developed for predicting unsteady loads on helicopter fuselages. An analogous problem for wind turbines is predicting loads on a turbine tower and nacelle, but those loads are of little engineering interest compared with the inertial loads of the spinning blades. However, since the actuator blade method captures the gross features of the turbine wake, it has potential for use in predicting interactions between multiple turbines in a wind farm. An upstream rotor with actuator blades (or its steady-state counterpart, the actuator disc) can be applied to predict the power loss experienced by a downstream turbine in its wake. Actuator blades could also approximate an unsteady upstream wake feeding into a high fidelity overset simulation downstream. In that manner, blade vortex interaction between a turbine and its upstream neighbors could be predicted. The principal difficulty in such a simulation would be the computational expense of running the simulation long enough to allow the wake to convect from one turbine to the next and the requirement of fine enough grid resolution between them to avoid dissipating the wake. This latter requirement can be alleviated by the application of adaptive grids [37, 114, 91] and/or higher order of accuracy algorithms [89, 61] in these regions.



(a) Axial position, z of vortices vs. vortex age.



(b) Radial position, r of vortices vs. vortex age.

Figure 10: Trajectories of root and tip vortices for both overset and actuator blades (with actual twist). The radial separation between the overset and actuator root vortices is approximately equal to the distance between the overset and actuator blade roots.

CHAPTER V

HYBRID RANS/LES

Even in zero-yaw states, wind turbine flowfields have very prominent unsteady, 3-D effects, including massive separations. These flow features cannot be accurately resolved using traditional RANS turbulence models. In this chapter, a hybrid RANS/LES turbulence model and its extension to an unstructured methodology is discussed. The model is then demonstrated for several configurations of interest.

5.1 Governing Equations of Motion

The instantaneous Navier-Stokes equations for compressible flows can be expressed in tensor form as:

$$\begin{aligned} \frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}(\rho u_j) &= 0 \\ \frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_j}(\rho u_i u_j) + \frac{\partial p}{\partial x_i} - \frac{\partial \sigma_{ji}}{\partial x_j} &= 0 \\ \frac{\partial}{\partial t}\left[\rho\left(e + \frac{1}{2}u_i u_i\right)\right] + \frac{\partial}{\partial x_j}\left[\rho u_j\left(h + \frac{1}{2}u_i u_i\right)\right] + \frac{\partial q_j}{\partial x_j} - \frac{\partial}{\partial x_j}(u_i \sigma_{ij}) &= 0 \end{aligned} \tag{6}$$

For flows relevant to common aerodynamic applications, several assumptions about the flow can be made to affect closure of these equations. The flows of interest are assumed to be in the moderate temperature range (incompressible to supersonic Mach regimes), comprised of a Newtonian fluid consisting of a monotonic gas with isotropic viscosity. Thus the perfect gas law and the Boussinesq constitutive relation can be applied.

The difference between RANS and LES first appears in the averaging technique applied to the Navier-Stokes equations. RANS equations apply Reynolds averaging where the primitive variables $(\rho, \vec{V}, p, e)$ are separated into mean $(\overline{f})$ and fluctuating (f') components, then averaged over a finite period of time much greater than the turbulent fluctuation frequency, so that the mean value of a single fluctuating variable will be zero. The mean value of some

multiplied fluctuating variables will remain non-zero, and these correlated values must be treated via additional assumptions. This Reynolds-averaging process is typically applied to the incompressible Navier-Stokes equations to obtain closure for the viscous turbulence. For LES simulations, density, heat transfer and pressure remain as Reynolds-averaged values, while the remainder of the pertinent variables are decomposed using a Favre-averaging that accounts for compressibility effects, and yields mass-averaged ($\widetilde{f}$) and fluctuating (f'') components. It should be noted that the fluctuating terms that arise in the Favre and Reynolds averaging processes are not identical and therefore are typically assigned different notations.

The compressible Navier-Stokes equations, once Favre-averaged, can be mathematically formulated as:

$$\begin{aligned}
\frac{\partial \overline{\rho}}{\partial t} + \frac{\partial}{\partial x_j}(\overline{\rho}\widetilde{u_j}) &= 0 \\
\frac{\partial}{\partial t}(\overline{\rho}\widetilde{u_i}) + \frac{\partial}{\partial x_j}(\overline{\rho}\widetilde{u_i}\widetilde{u_j}) &= -\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial \overline{\sigma_{ij}}}{\partial x_j} - \frac{\partial}{\partial x_j}\left(\overline{\rho u_i'' u_j''}\right) \\
\frac{\partial}{\partial t}\left[\overline{\rho}\left(\widetilde{e} + \frac{1}{2}\widetilde{u_i}\widetilde{u_i}\right)\right] + \frac{\partial}{\partial x_j}\left[\overline{\rho}\widetilde{u_j}\left(\widetilde{h} + \frac{1}{2}\widetilde{u_i}\widetilde{u_i}\right)\right] &= \frac{\partial}{\partial x_j}(-q_{L_j} + \widetilde{u_i}\overline{\sigma_{ij}}) - \frac{\partial}{\partial t}\left(\frac{1}{2}\overline{\rho u_i'' u_i''}\right) \\
&\quad + \frac{\partial}{\partial x_j}\left(-\widetilde{u_j}\overline{\rho u_i'' u_j''} + \widetilde{u_j}\frac{1}{2}\overline{\rho u_i'' u_i''} - \overline{\rho u_j'' h''}\right. \\
&\quad \left. + \overline{\sigma_{ji} u_i''} - \frac{1}{2}\overline{\rho u_j'' u_i'' u_i''}\right) \qquad (7)
\end{aligned}$$

As a result of the averaging processes for incompressible or compressible assumptions, there arises a Reynolds-stress tensor that requires closure, either in the specific Reynolds-averaged form ($\tau_{ij} = -\overline{u_i' u_i'}$) or the compressible Favre-averaged form ($\overline{\rho}\tau_{ij} = -\overline{\rho u_i'' u_i''}$). The turbulent kinetic energy can then be defined in its specific formulation $k = \frac{1}{2}\overline{u_i' u_i'}$ or conserved form $\overline{\rho}k = \frac{1}{2}\overline{\rho u_i'' u_i''}$, respectively. Favre-averaging also gives rise to the turbulent heat flux ($q_{Ti} = \overline{\rho u_i'' h''}$) and the rate of turbulent dissipation ($\overline{\rho}\epsilon = \overline{\sigma_{ji}\frac{\partial u_i''}{\partial x_j}}$). For flows up through the low supersonic regime, the molecular diffusion and turbulent transport terms ($\overline{\sigma_{ji} u_i''} - \overline{\rho u_j'' \frac{1}{2} u_i'' u_i''}$) are typically ignored [113], and this practice is continued for this analysis. The additional equations through which these terms are resolved or closed gives rise to the turbulence model of the simulation:

$$\frac{\partial \overline{\rho}}{\partial t} + \frac{\partial}{\partial x_j}(\overline{\rho}\widetilde{u_j}) = 0$$

$$\frac{\partial}{\partial t}(\overline{\rho}\widetilde{u_i}) + \frac{\partial}{\partial x_j}(\overline{\rho}\widetilde{u_i}\widetilde{u_j}) = -\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial \overline{\sigma_{ij}}}{\partial x_j} - \frac{\partial}{\partial x_j}(\overline{\rho}\tau_{ij})$$

$$\frac{\partial}{\partial t}\left[\overline{\rho}\left(\widetilde{e} + \frac{1}{2}\widetilde{u_i}\widetilde{u_i}\right)\right] + \frac{\partial}{\partial x_j}\left[\overline{\rho}\widetilde{u_j}\left(\widetilde{h} + \frac{1}{2}\widetilde{u_i}\widetilde{u_i}\right)\right] = \frac{\partial}{\partial x_j}(-q_{L_j} + \widetilde{u_i}\overline{\sigma_{ij}}) - \frac{\partial}{\partial t}\left(\overline{\rho}k\right) + \frac{\partial}{\partial x_j}\left(-\widetilde{u_j}\overline{\rho}\tau_{ij} + \widetilde{u_j}\overline{\rho}k - q_{T_j}\right) \quad (8)$$

If symmetry is assumed, the Reynolds-stress tensor yields six unknowns that are approximated using models about the behavior of the fluctuating correlations, $\overline{u_i'' u_j''}$. These approximations yield the set of RANS turbulence models, ranging from algebraic to two-equation techniques. It was previously noted that the current practice is to assume the Boussinesq approximation, which can be utilized to relate the fluctuations to an eddy viscosity, μ_T:

$$\overline{\rho}\tau_{ij} = 2\mu_T\left[S_{ij} - \frac{1}{3}\frac{\partial \widetilde{u_k}}{\partial x_k}\delta_{ij}\right] - \frac{2}{3}\overline{\rho}k\delta_{ij} \quad (9)$$

Similarly, the turbulent heat flux vector can be related to the eddy viscosity, μ_T, via proportionality to the mean temperature gradient:

$$q_{Ti} = -\frac{\mu_T}{Pr_T}\frac{\partial \widetilde{h}}{\partial \widetilde{x_j}} = -\frac{\mu_T c_p}{Pr_T}\frac{\partial \widetilde{T}}{\partial \widetilde{x_j}} \quad (10)$$

that introduces the turbulent Prandtl number, Pr_T, which can be either constant or variable, depending on the application. Finally, the rate of turbulent dissipation can be expressed as

$$\overline{\rho}\epsilon = \overline{\mu\left[2S_{ji}S_{ij}'' - \frac{2}{3}u_{kk}u_{ii}''\right]} \quad (11)$$

In addition to the Favre-averaging, the concept of LES is based on the direct capture of the large turbulence eddies as part of the solution of the Favre-averaged Navier-Stokes equations, relegating the smaller turbulent eddies to be modeled. This process is based on the view that the larger turbulence eddies contribute significantly to the Reynolds-stress tensor, while the smaller eddies are less significant. In order to separate these effects, in addition to the averaging process, the variables in the equation of motion should also be filtered (typically referred to as Favre-filtering) to obtain the small or subgrid scale (sgs)

turbulence. An excellent discussion of these filtering techniques can be found in Wilcox [113]. Near the surface of the configuration undergoing simulation, the turbulent eddy scales reduce significantly, requiring in LES a very refined grid that increases the computational resources beyond the reach of most engineering applications. As attached boundary layer characteristics can be well-predicted by RANS turbulence models, an alternative to grid refinement is to switch the simulation between RANS and LES.

5.2 Hybrid RANS-LES Model

The correlation of these turbulence terms requiring closure to viscosity permits the RANS and LES turbulence approximations to be combined to provide closure information in the simulation. The information exchange occurs via the turbulent kinetic energy, k. In this work, the RANS turbulence model chosen to effect this closure is the Menter k-ω SST turbulence model [55], based on its success in prior CFD applications, for example [93, 88, 78].

The Menter k-ω SST turbulence model resolves two differential equations that describe the turbulent kinetic energy, k, as well as an approximation for the length scale based on the dissipation per unit turbulent kinetic energy, ω. These equations are given by:

$$\frac{\partial}{\partial t}(\rho k) + \frac{\partial}{\partial x_j}(\rho u_j k) = \tau_{ij}^{rans}\frac{\partial u_i}{\partial x_j} - \beta^*\rho\omega k + \frac{\partial}{\partial x_j}\left[(\mu + \sigma_k\mu_T)\frac{\partial k}{\partial x_j}\right] \tag{12}$$

$$\begin{aligned}\frac{\partial}{\partial t}(\rho\omega) + \frac{\partial}{\partial x_j}(\rho u_j \omega) = {} & \frac{\gamma\rho}{\mu_T}\tau_{ij}^{rans}\frac{\partial u_i}{\partial x_j} - \beta\rho\omega^2 + \frac{\partial}{\partial x_j}\left[(\mu + \sigma_\omega\mu_T)\frac{\partial\omega}{\partial x_j}\right] \\ & + 2(1 - F_2)\rho\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial\omega}{\partial x_j}\end{aligned} \tag{13}$$

where the *rans* superscript is used to denote the use of the Reynolds-averaged Reynolds-stress tensor. Menter [55] indicates that the production terms ($\tau_{ij}^{rans}\frac{\partial u_i}{\partial x_j}$) can be modeled directly with $\mu_T\Omega^2$ where Ω is the vorticity magnitude defined by $\Omega^2 = (w_y - v_z)^2 + (u_z - w_x)^2 + (v_x - u_y)^2$.

The LES turbulent kinetic energy equation to obtain the subgrid scale data is one successfully used by Kim and Menon [39]:

$$\frac{\partial}{\partial t}(\overline{\rho}k^{sgs}) + \frac{\partial}{\partial x_j}(\overline{\rho}\widetilde{u_j}k^{sgs}) = \tau_{ij}^{sgs}\frac{\partial\widetilde{u_i}}{\partial x_j} - C_\epsilon\overline{\rho}\frac{(k^{sgs})^{3/2}}{\Delta} + \frac{\partial}{\partial x_j}\left[\left(\frac{\widetilde{\mu}}{Pr} + \frac{\mu_{sgs}}{Pr_t}\right)\frac{\partial k^{sgs}}{\partial x_j}\right] \tag{14}$$

Baurle et al. [12] have demonstrated that RANS and LES methods can be linearly merged to form a hybrid model. Speziale [101] proposed an extension to this via the Reynolds-stress tensor. Thus the RANS equations of motion and kinetic energy equation would be recast in the generic form:

$$\frac{\partial}{\partial t}(\vec{E}) + \frac{\partial}{\partial x_j}(\widetilde{u_j}\vec{E}) = \frac{\partial}{\partial x_j}(\vec{G}^{trans}) + \vec{G}^{src} + \frac{\partial}{\partial x_j}(\vec{G}^{hybrid}_{T_{trans}}) + \vec{G}^{hybrid}_{T_{src}} \tag{15}$$

where $\vec{E} = \{\rho, \rho\widetilde{u_j}, \rho E, \rho k\}$, and the right hand side of the equation consists of the original transport ($\vec{G}^{trans}$) and source ($\vec{G}^{src}$) vectors excluding the fluctuating turbulence terms, which have been formulated into new vectors, $\vec{G}_{T_{trans}}$ and $\vec{G}_{T_{src}}$ that will be hybridized. The hybridization of the two $\vec{G}_T$ terms occurs via a simple linear formulation:

$$\vec{G}^{hybrid}_T = F\vec{G}^{rans}_T + (1-F)\vec{G}^{sgs}_T. \tag{16}$$

The function F which is used as the switching mechanism is:

$$F = \tanh(x^4) \text{ where } x = \max\left(\frac{2\sqrt{k}}{0.09\omega y}, \frac{500\nu}{y^2\omega}\right). \tag{17}$$

Further details of the development of this hybridization technique can be found in publications by Sanchez-Rocha and Menon [79, 80].

The first term on the right-hand side of Eq. 14 is the k^{sgs} production term. If this term is not included in the hybridization, so that all production of turbulent kinetic energy comes from the corresponding term in Eq. 12, the resulting model is very similar to a DES model. In a DES model, the underlying RANS model functions as an SGS model in regions of the grid with sufficient resolution. The RANS destruction terms are simply modified to depend on some characteristic length scale [109]. The fact that the model discussed here is hybridized with a separate SGS model means that it is not a DES model. However, if the choice is made not to hybridize production, the hybridized destruction term behaves similarly to the destruction term within a DES model based on the k-ω SST RANS model [113]. Henceforth, the term hybrid RANS-LES (HR-LES) will refer to the full hybridization, while hybrid RANS-DES (HR-DES) will be used to refer to a form where production is not hybridized.

5.3 Unstructured Implementation

When implementing the HRLES scheme into any existing code, it is necessary to nondimensionalize the turbulence model equations in accordance with the code conventions. For FUN3D, the nondimensionalization consists of the following:

$$\rho = \frac{\hat{\rho}}{\hat{\rho}_\infty}, \quad u_j = \frac{\hat{u}_j}{\hat{a}_\infty}, \quad k = \frac{\hat{k}}{\hat{a}_\infty^2}, \quad \omega = \frac{\hat{\mu}_\infty \hat{\omega}}{\hat{\rho}_\infty \hat{a}_\infty^2}, \quad x_j = \frac{\hat{x}_j}{\hat{L}}, \quad t = \frac{\hat{t}}{\hat{L}/\hat{a}_\infty}$$

$$\frac{\partial}{\partial \hat{x}_j} = \frac{\partial}{\partial x_j}\frac{\partial x_j}{\partial \hat{x}_j} = \frac{1}{\hat{L}}\frac{\partial}{\partial x}, \qquad \frac{\partial}{\partial \hat{t}} = \frac{\partial}{\partial t}\frac{\partial t}{\partial \hat{t}} = \frac{\hat{a}_\infty}{\hat{L}}\frac{\partial}{\partial t} \tag{18}$$

where the caret symbol represents a dimensional variable. The effect of this nondimensionalization on the governing equations is illustrated below with the k and ω turbulence equations. The Boussinesq approximation of Eq. 9 was used to form the production term in Eq. 20.

$$\frac{\partial}{\partial t}(\rho k^{rans}) + \frac{\partial}{\partial x_j}(\rho u_j k^{rans}) = \frac{M_\infty}{Re}\frac{\mu_T}{\rho}\Omega^2 - \frac{Re}{M_\infty}\beta^*\rho\omega k^{rans} + \frac{\partial}{\partial x_j}\left[(\mu + \sigma_k \mu_T)\frac{\partial k}{\partial x_j}\right] \tag{19}$$

$$\begin{aligned}\frac{\partial}{\partial t}(\rho k^{sgs}) + \frac{\partial}{\partial x_j}(\rho u_j k^{sgs}) = &\left[\frac{2\mu_T}{\rho}\frac{M_\infty}{Re}\left(S_{ij} - \frac{1}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right) - \frac{2}{3}k^{sgs}\delta_{ij}\right]\frac{\partial u_i}{\partial x_j} \\ &+ C_\epsilon \rho \frac{k^{sgs3/2}}{\Delta} + \frac{\partial}{\partial x_j}\left[\left(\frac{\mu}{Pr} + \frac{\mu^{sgs}}{Pr_t}\right)\frac{\partial k^{sgs}}{\partial x_j}\right]\end{aligned} \tag{20}$$

$$\begin{aligned}\frac{\partial}{\partial t}(\rho\omega) + \frac{\partial}{\partial x_j}(\rho u_j \omega) = &\frac{M_\infty}{Re}\gamma\rho\Omega^2 - \frac{Re}{M_\infty}\beta\rho\omega^2 + 2(1-F_2)\frac{Re}{M_\infty}\rho\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j} \\ &+ \frac{\partial}{\partial x_j}\left[(\mu + \sigma_\omega \mu_T)\frac{\partial \omega}{\partial x_j}\right]\end{aligned} \tag{21}$$

In a node-centered framework, the boundary condition for turbulent kinetic energy at solid walls is simply $k = k^{sgs} = 0$. Since the dissipation equation is not blended, the boundary condition on ω is that of the k-ω SST model. With the terms non-dimensionalized as above, this is expressed as:

$$\omega_{wall} = \left(\frac{M_\infty}{Re}\right)^2 \frac{60\nu}{\beta_1 y^2}, \tag{22}$$

where y is the distance from the nearest solid wall. Since this distance is zero along a viscous boundary, it is instead set to the normal distance from the wall of the nearest node adjacent to the wall.

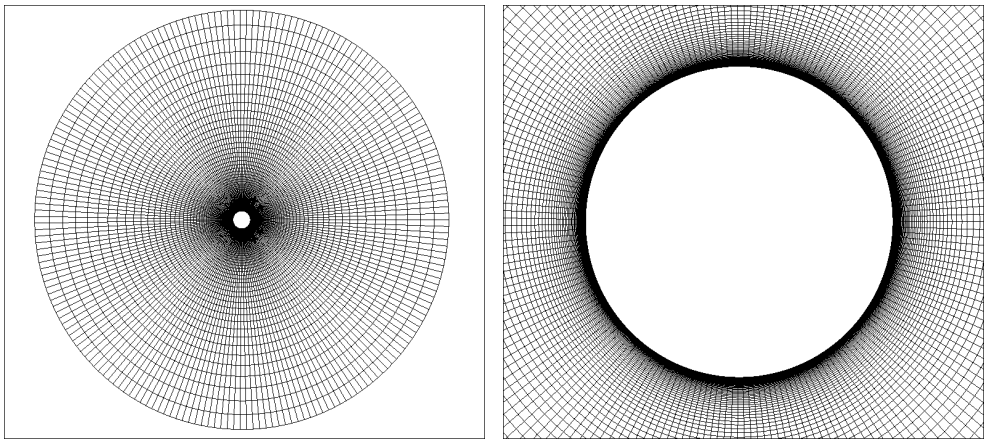
5.4 Method Verification

Evaluations of a circular cylinder were utilized to verify the new turbulence method and examine its impact on highly separated flows. The first series of evaluations compared the ability of the unstructured hybrid RANS-LES method to capture the flow physics with respect to RANS, structured hybrid RANS-LES, and LES results. Additional evaluations applied the model to grid systems of differing element topologies to determine the behavior of the model on these mixed topology meshes.

5.4.1 Initial Evaluation: Circular Cylinder

The initial evaluation of the hybrid RANS-LES method simulates the flow of air over a circular cylinder at a Mach number of 0.2 and a diameter-based Reynolds number of 3900 for standard sea-level conditions. The k-ω SST and HR-DES models (non-hybridized production terms) were used to simulate this case. The grid about each cylindrical section is a structured O-mesh that was converted to form an unstructured hexahedral grid. The grid included 200 nodes in the wrap-around direction with 139 points expanding radially outward with a 10% stretching ratio, yielding a $y^+ < 0.02$. While the wake was somewhat coarse due to the uniform spacing of the grid as seen in Fig. 11, it was instrumental in verifying the implementation of the model, allowing a direct comparison with the structured methodology implementations. A nondimensional time step of 0.119 was selected to obtain approximately 200 steps per shedding cycle, assuming a Strouhal number of 0.21.

Figure 12 shows instantaneous contours of vorticity magnitude about the 2-D circular cylinder using RANS (k-ω SST) and HRLES models. Even in two dimensions, HRLES captures the physics associated with the vortex shedding much more accurately when compared to the baseline k-ω SST RANS results that have been run on an identical grid and with the same numerical options. The RANS solution shows that the vortex shedding process has been completely smoothed out by the model, while the vortex wake of the hybrid RANS-LES

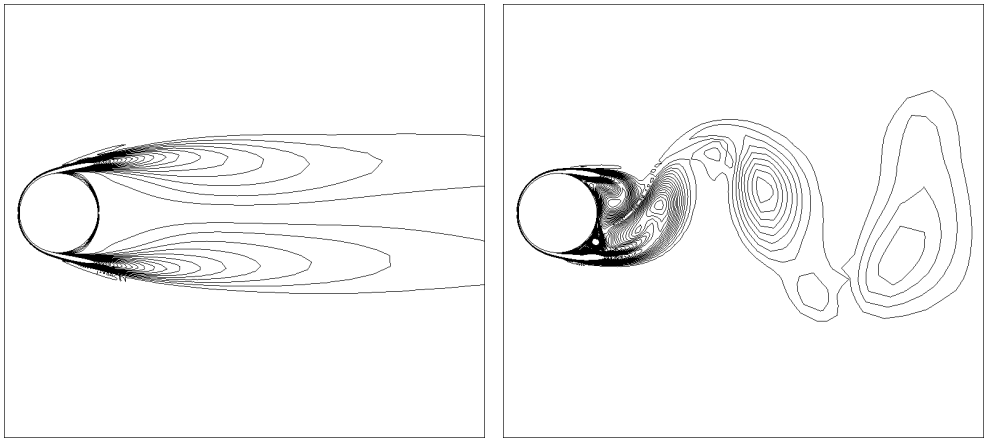


(a) Complete grid (b) Close to surface

Figure 11: Cross-sectional grid for the cylinder verification simulation.

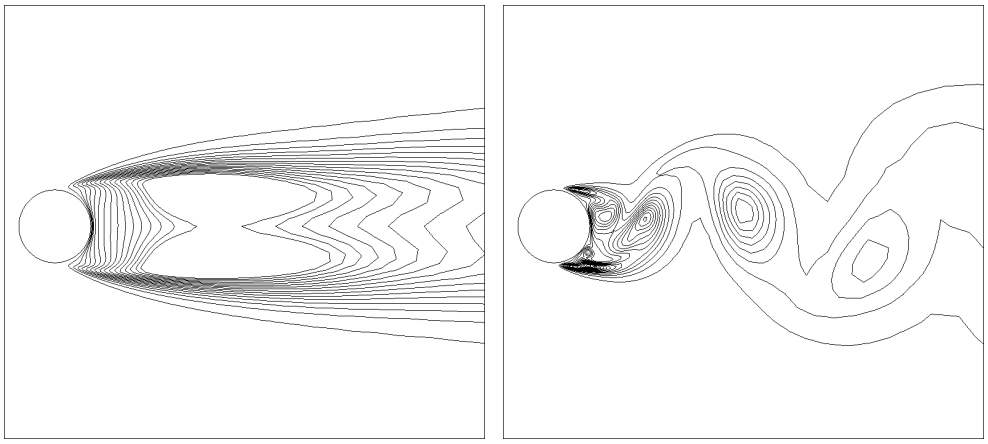
solution includes features that are more consistent with the physical assumptions. This is borne out when the values of the turbulent kinetic energy are observed, as in Fig. 13, where the turbulent kinetic energy shows a smeared wake region with no periodicity. Figure 14 is a graphical representation demonstrating the blending of the two models. Here, values near zero indicate LES-dominated regions and values near one indicate RANS-dominant regions. Near the surface and attached regions, the blending function is close to one, indicating that the RANS model is dominant. In the wake and aft of the separation point, the function approaches zero, denoting that the flow field turbulence characteristics are dominated by the LES-computed turbulence.

The computed Strouhal number for the 2-D hybrid RANS-LES simulation is 0.25, compared with the experimental [70] value of 0.215 ± 0.005. The RANS simulation did not shed vorticity, so it was not possible to compute a Strouhal number. The separation point predicted by the hybrid method is 86.6°, while the RANS method predicts separation at 85.1°, both close to the experimentally determined location of $86.0° \pm 2°$, which was extracted from data obtained at $Re_D = 5000$ [95]. The drag coefficient for hybrid RANS-LES simulation is computed to be 1.5, which is significantly higher than the experimental[40] value of 0.99 ± 0.05, but is in line with the value of 1.65 obtained by similar LES simulations[40] computed also in two dimensions. The k-ω SST simulation resulted in a drag coefficient of



(a) k-ω SST (b) Hybrid RANS-LES

Figure 12: Vorticity magnitude about the 2-D circular cylinder.



(a) k-ω SST (b) Hybrid RANS-LES

Figure 13: Turbulent kinetic energy about the 2-D circular cylinder.

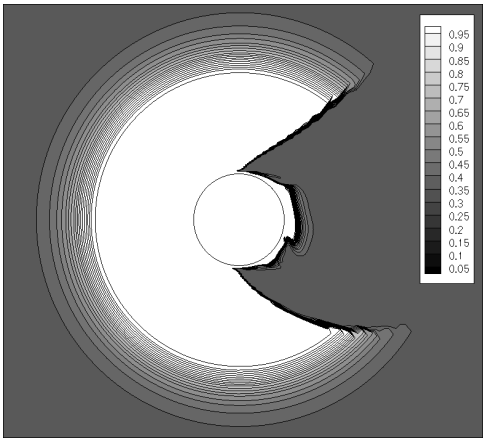


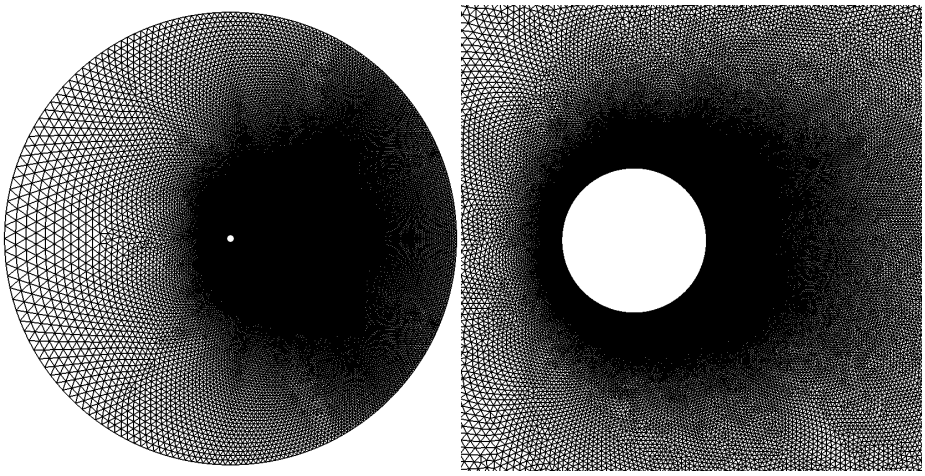
Figure 14: Instantaneous contours of the blending function of Eq. 17 for the 2-D circular cylinder. Lighter areas are RANS-dominated, and darker areas are LES-dominated.

0.887.

5.4.2 Unstructured Mesh Studies

A second set of circular cylinder evaluations was evaluated on three different grid systems. These evaluations use fully unstructured grids that were generated as a two-dimensional mesh of triangle elements and quadrilateral elements (the latter in the boundary layer) and then extruded in the spanwise direction to form prismatic elements. This strategy permits a straightforward variation of the spanwise resolution, which has been demonstrated [94] to be an important factor in accurate CFD solutions of separated flows. The first grid had 51 planes spaced evenly over a 4-diameter span, the second had 101 planes over 4 diameters, and the third had 101 planes over 8 diameters, yielding the same number of nodes as the second grid, but with larger cell aspect ratios. A radial cross-section of the grid, shown in Fig. 15, contains 123,652 nodes, 203,492 triangular elements, and 21,363 quadrilateral elements. Normal spacing was such that $y^+ << 1$. A refined nondimensional time step of $\Delta t a_\infty / D = 0.025$ provided approximately 1000 time steps per vortex shedding cycle. In these simulations, the fully hybridized HR-LES model was used.

While hybrid RANS-LES was earlier shown to yield some improvement over k-ω SST in characterization of the flowfield about a 2-D cylinder, the improvement is even more significant in a 3-D simulation. Figure 16 presents iso-surfaces of the vortex identification criterion, sometimes called *Q criterion*, for the 3-D cylinder at $Re = 3900$ on the three grids



(a) (b)

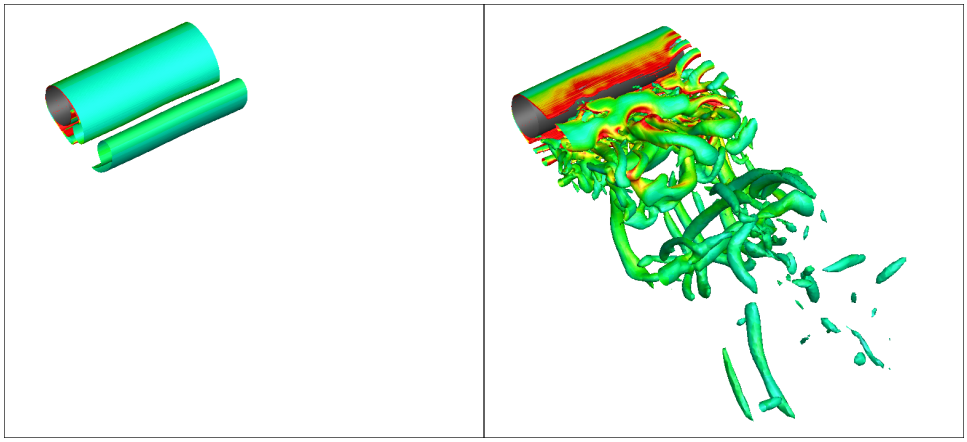
Figure 15: Computational grid for the 3-D cylinder.

described in section 5.4.2. The Q criterion is defined as:

$$Q = \frac{1}{4}||\vec{\Omega}||^2 - \left[\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial u}{\partial z}\right)^2 + \frac{1}{2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)^2 + \frac{1}{2}\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right)^2 + \frac{1}{2}\left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z}\right)^2\right]. \quad (23)$$

The definition of Q is such that it is only positive near vortices. In flow visualizations, this property can be used to filter out the "sheets" of high vorticity that emanate from turbulent boundary layers, revealing vortices. While the k-ω SST model captures periodic vortex shedding in three dimensions where it did not in two, those vortices are essentially 2-D across the span (Fig. 16(a)). In contrast, the hybrid RANS-LES results on all grids show substantial spanwise variation and a much more irregular wake (Fig. 16(b-d)). Since k-ω SST yields a result that could have been obtained on a substantially reduced grid (whatever the method requires for a 2-D simulation), the majority of the grid nodes have simply added to the overall cost of the computation.

Table 1 delineates the various statistics for the cylinder predictions, including mean drag coefficient, Strouhal number, and separation location. Strouhal number is calculated from the frequency spectrum of the fluctuating lift. Separation location is given in degrees over the circumference of the cylinder from the leading edge stagnation point to the point where skin friction along the cylinder centerline drops to zero. All 3-D cases predict the



(a) k-ω SST, 4 diameters, 101 planes

(b) HR-LES, 4 diameters, 51 planes

(c) HR-LES, 4 diameters, 101 planes

(d) HR-LES, 8 diameters, 101 planes

Figure 16: Iso-surfaces of Q criterion about a 3-D circular cylinder with varying grid resolution.

Table 1: Predicted circular cylinder characteristics for various turbulence methods and grids with number of spanwise planes, N_z, and spanwise extent, Z. Separation location is given in degrees of azimuth from the leading edge stagnation point. Experimental Strouhal number is from Reference [70], and separation location is from Reference [95] at $Re_D = 5000$. LES data is from Reference [40].

N_z	Z	Turbulence model	Mean C_D	Strouhal no.	Separation location
–	–	Exp.	0.99 ± 0.05	0.215 ± 0.005	$86 \pm 2°$
101	$4D$	k-ω SST	1.456	0.213	98.4°
2	–	HR-LES	1.5	0.25	86.6°
51	$4D$	HR-LES	0.971	0.216	85.8°
101	$4D$	HR-LES	0.919	0.217	84.3°
101	$8D$	HR-LES	0.939	0.217	84.7°
48	πD	LES	1.04	0.210	88.0°

Strouhal number within experimental bounds, regardless of turbulence model. However, the 3-D k-ω SST simulation yields a drag coefficient over 1.4, close to the result of a 2-D HR-LES simulation. With 3-D HR-LES, drag is predicted much closer to the experimental value, though it is still slightly under-predicted on all grids. All HR-LES simulations predict the separation location within the experimental bounds, while k-ω SST predicts it over 10° farther aft.

Figure 17 shows mean pressure coefficient along the cylinder centerline, from the leading edge stagnation point ($\theta = 0$) to 180° opposite. There, it is clear that the HR-LES simulations better predict drag because they do not over-predict base suction to the extent that the k-ω SST simulation does. Figure 18 shows the correlation between computed and measured centerline C_p, with any points on a diagonal line with a 45° slope being in perfect agreement with the measured data. The zoomed in region in Figure 18(b) indicates that the grid with a diameter of $4D$ and 51 spanwise planes has the best correlation once the flow is separated.

Unlike their structured grid counterparts, which only allow hexahedral cells, unstructured CFD solvers may allow a variety of cell types. In structured formulations, very high cell aspect ratios are quite common in directions where flow gradients are expected to be small. However, in unstructured frameworks, tetrahedral cells of very high aspect ratio, which are commonly seen in boundary layers, pose special problems. Figures 19(a) and (b)

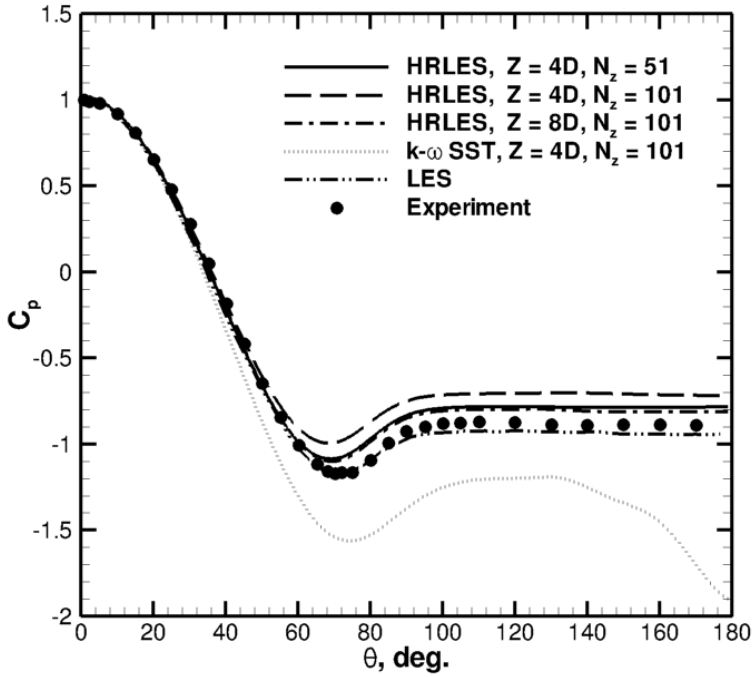
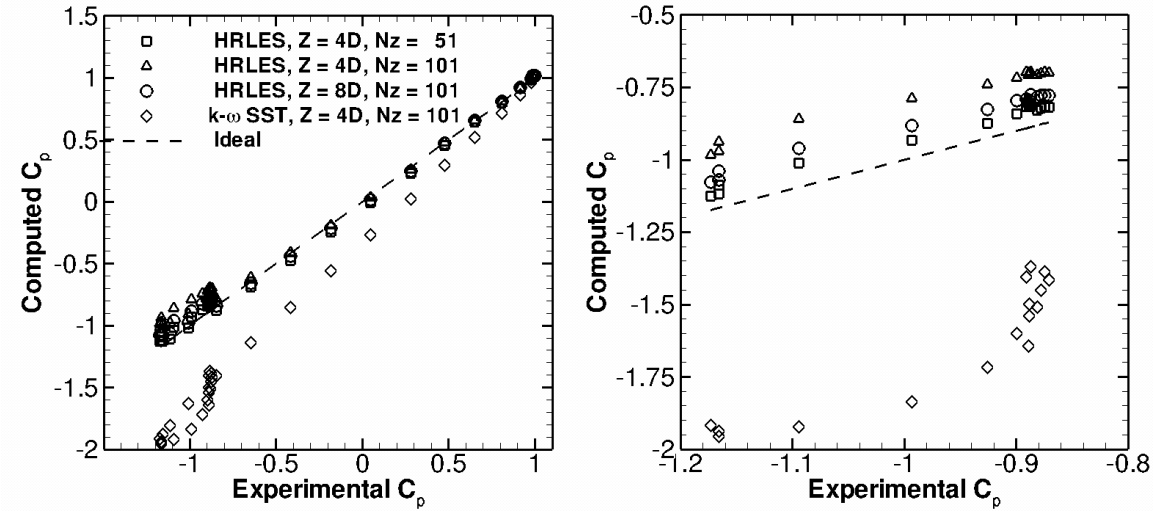


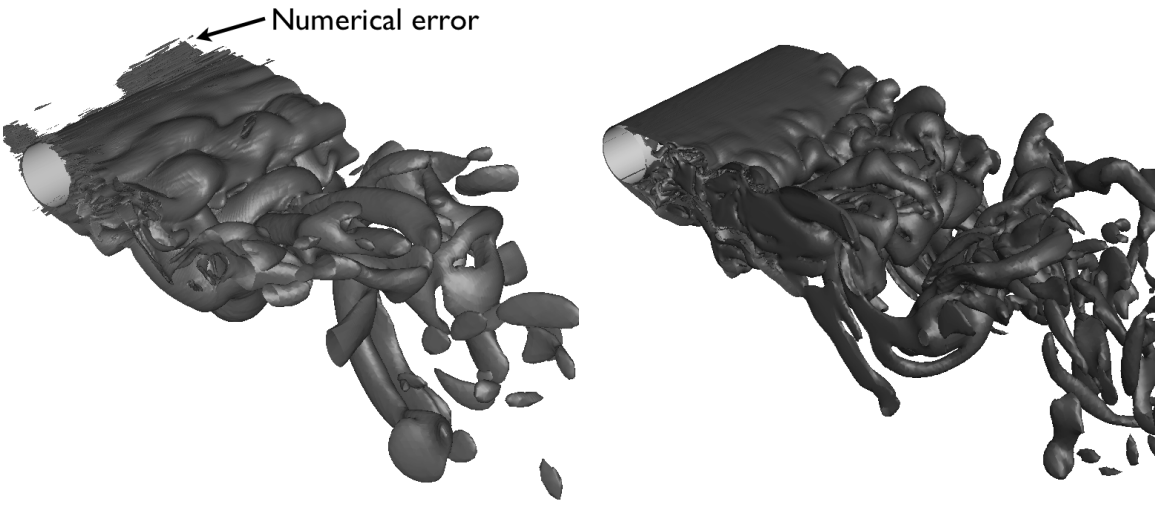
Figure 17: Mean pressure coefficient along the centerline for the 3-D circular cylinder on several grids with both RANS and HR-LES turbulence models. LES data is from Reference [40], and experimental data is from Reference [65].



(a) All data points

(b) Points with $\theta > 70°$

Figure 18: Correlation of computed mean pressure coefficient with measured value.



(a) Tetrahedral boundary layer cells (b) Prismatic boundary layer cells

Figure 19: Iso-vorticity contours around a 3-D cylinder, computed using the HR-LES model on grids with different cell types in the boundary layer.

depict iso-surfaces of vorticity magnitude on the 3-D cylinder using tetrahedral and prismatic boundary layer cells. The tetrahedral boundary layer solution indicates the presence of numerical errors in the form of spurious vorticity upstream of the leading edge. These errors are not present when the boundary layer is modeled with prismatic cells.

Node-centered solvers typically use an edge-based scheme in which fluxes are evaluated along the edges of the primal grid. Since each primal edge corresponds to a face in the dual mesh, this is approximately equivalent to evaluating fluxes across the dual faces. In highly stretched tetrahedral cells, some dual faces and primal edges will be nearly parallel, yielding a very poor approximation of gradients across those faces. In turn, this reduces the accuracy of the reconstruction scheme, and errors propagate through the solution. Therefore, when applying this hybrid RANS-LES model, care should be taken to use prismatic cells in the boundary layer.

Profiles of the streamwise velocity, u/U_∞, at seven downstream wake traverse locations on the mesh with 101 planes over four diameters are shown in Figure 20. These profiles have been averaged in time and in space, in the spanwise direction. Experimental data from Reference [48] as well as LES data of Kravchenko and Moin [40] is also included. In the far wake, HR-LES slightly over-predicts the velocity deficit, but captures the spreading

of the wake observed in the experimental data. At the position closest to the wall, at $x/D = 0.58$, HR-LES predicts the correct trough-like profile. At the first position in the recirculation region, $x/D = 1.06$, HR-LES predicts a velocity profile that looks similar to those closer to the cylinder, continuing to produce a relatively thin shear layer. In general, in the recirculation region, the HR-LES profile "lags" in space behind the experimental and LES data, with the profile spreading farther downstream. From Table 1, it is clear that these discrepancies in the wake do not lead to large errors in drag or shedding frequency.

The errors in Figure 20 can be attributed in some part to grid topology. As described earlier, the mesh was originally generated in two dimensions and then extruded in the spanwise direction, which results in cells that have aspect ratios as high as 6 in the wake outside the boundary layer and as high as 40 in the farfield. The hybrid RANS-LES destruction term uses (cell volume)$^{1/3}$ as a measure of the local length scale, which implicitly assumes a nearly isotropic grid. This topology consideration is distinct from the earlier discussion of boundary layer cell type in that the issue is not one of gradient reconstruction, but rather one of reasonably estimating the length scale representing the largest eddy that can be resolved on the grid.

To test the effect of grid topology, a naturally-developing 3-D grid was created. Instead of a defined number of spanwise points, the cells were allowed to expand in the spanwise direction at the same rate as the other two directions. So while the surface resolution was similar to the original mesh, the total node count was substantially lower, only 5 million. Figure 21 shows a spanwise slice of the two grids near the four upstream traverses. Wake velocity profiles at the seven traverses are shown again in Figure 22. The $x/D = 1.54$ and 2.02 locations show substantial improvement over the RANS predictions, with the former attaining more of the expected V-shaped profile, though still spreading slower than expected. Subtle improvements are visible at other traverse locations as well.

Another potential cause of the discrepancies in the velocity profiles in Figures 20 and 22 is the sub-critical Reynolds number of these simulations. $Re_D = 3900$ was chosen to allow correlation of HR-LES results with the LES work of Kravchenko and Moin [40], but at that Reynolds number, all separation on the cylinder is expected to be be laminar. Transition

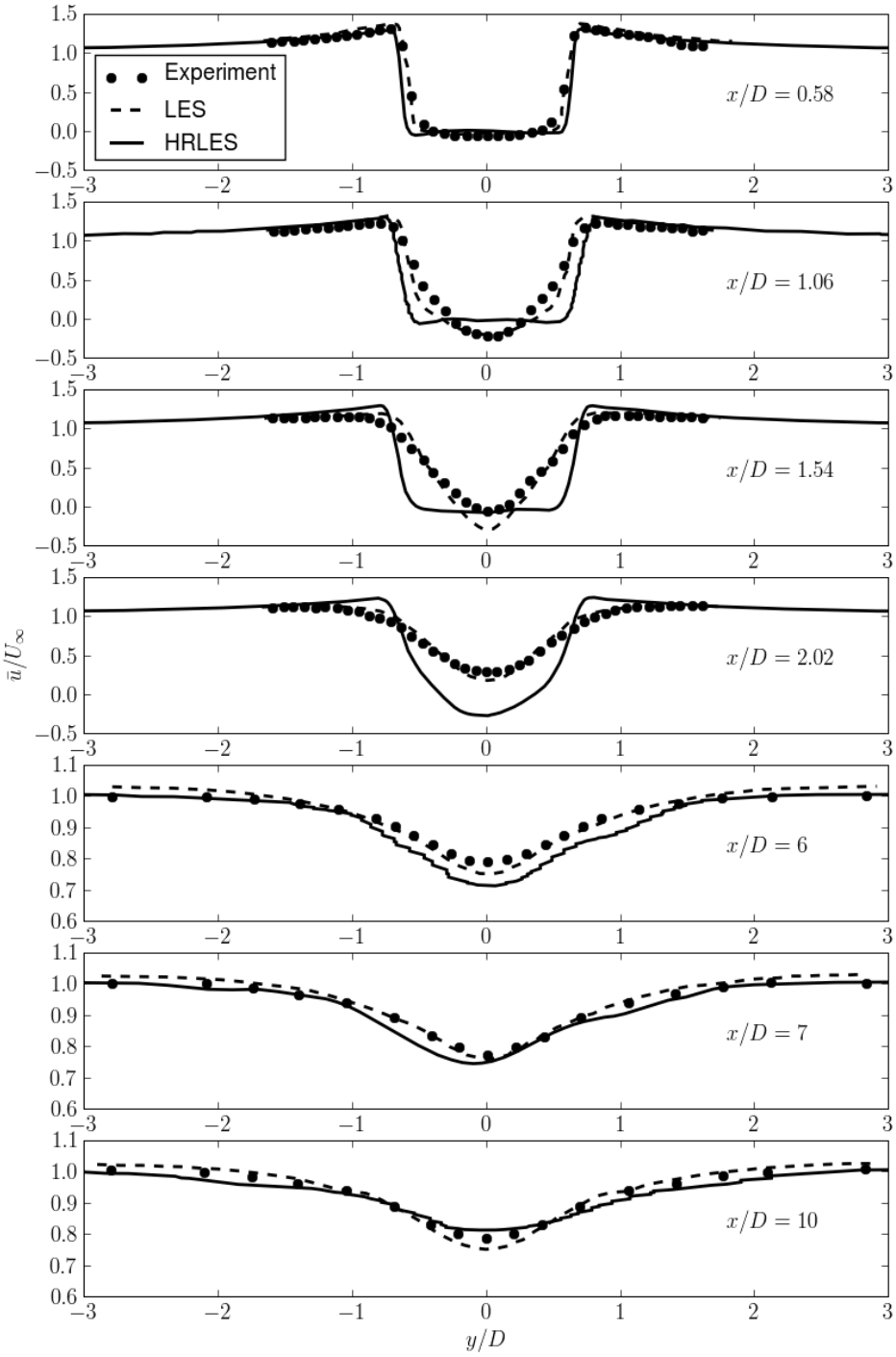
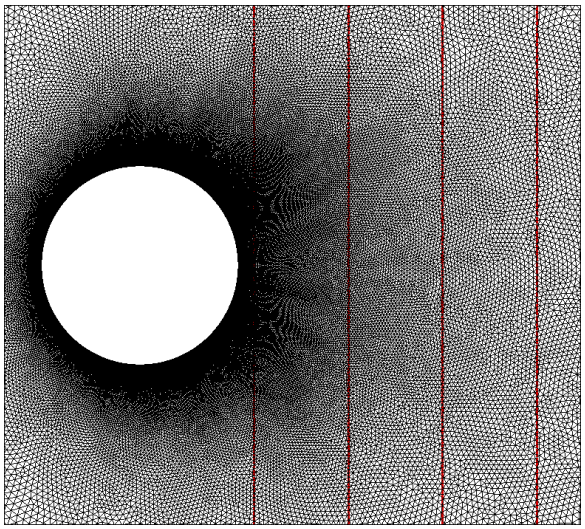
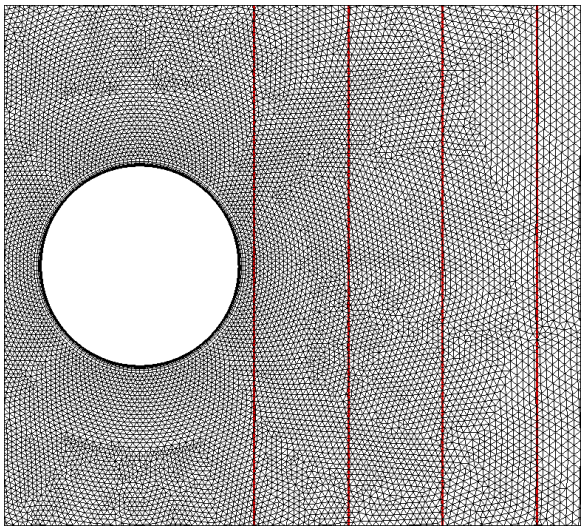


Figure 20: u/U_∞ at seven locations in the wake of a cylinder at $Re = 3900$ using a spanwise extruded grid.



(a) Extruded mesh, 101 spanwise planes over 4 diameters



(b) Truly 3-D mesh

Figure 21: Spanwise slices of the original mesh and the 3-D mesh used to evaluate the effect of cell stretching. Red lines indicate wake traverse locations.

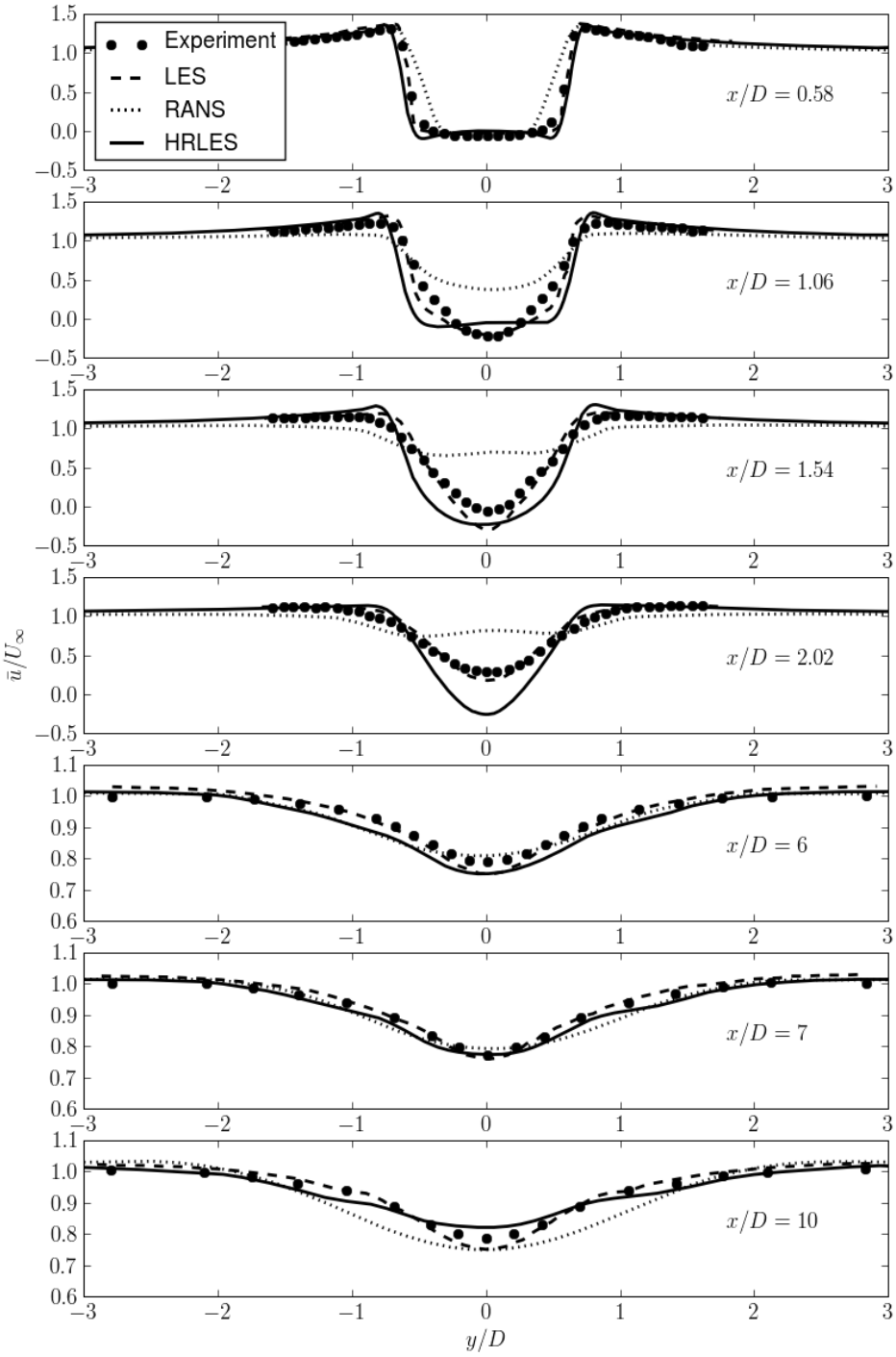


Figure 22: u/U_∞ at seven locations in the wake of a cylinder at $Re = 3900$ using a fully 3-D grid.

actually occurs in the shear layers separating the recirculation region from the outer flow [81]. Yet, it is assumed that the flow is fully turbulent in these simulations. A dedicated LES solver would better predict transition, provided the grid was fine enough to resolve the instability waves. Those waves would then propagate upstream an influence the flow near the wall. With a hybrid RANS-LES model, some improvement can be expected since due to the elliptical nature of the governing equations at this Mach number, information from the wake can propagate upstream. However, since the RANS model is still used at the wall, instabilities captured in the wake by the LES model will be averaged by the RANS model at the wall. As a result, HRLES methods still require a dedicated transition model to achieve the best results at sub-critical Reynolds numbers. Furthermore, Kravchenko and Moin [40] note that this case is very sensitive to disturbances such as the freestream turbulence level. They do not specify the freestream turbulence intensity in their LES simuations, but in the present HRLES simulations, it was set at $\sqrt{k}/U_\infty = 4.7 \times 10^{-4}$, which may be lower than the turbulence level in the experimental results used for correlation.

Finally, the coefficients of the LES portion of the method examined in this work are constant. Sanchez-Rocha [80] found that using constant coefficients could lead to artificial turbulence dissipation. If the wake turbulence is dissipated, there is less mixing across shear layers, possibly leading to the abrupt, square velocity profile seen at $x/D = 1.54$ in Figure 20. These results may be improved using a fully dynamic implementation where the coefficients are varied based on the local flow characteristics [80].

CHAPTER VI

FLATBACK AIRFOIL APPLICATIONS

As new wind turbine rotors get larger, novel techniques, including lightweight composite materials, are required to reduce stress on the blades and drivetrain. Flatback airfoils, which have very blunt trailing edges have been proposed for this purpose because their thick cross-section can bear those loads with less structural material [71]. However, due to their blunt cross-section, increased strength comes at the price of increased drag [102].

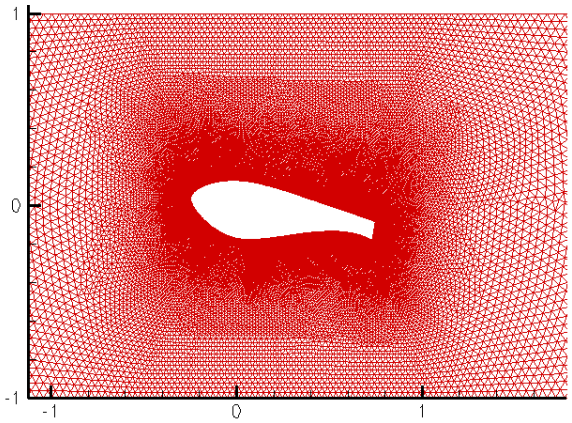
The hybrid RANS-LES model was used to model the flow over a DU97 flatback airfoil, which includes a 10%c thick trailing edge. The simulagion conditions were set to mirror, as closely as possible, wind tunnel data obtained for this airfoil by Berg and Zayas [14]. All cases were simulated using a compressible formulation with a Reynolds number of 3 million based on chord and a Mach number of 0.2. The latter was chosen to be low enough that compressibility effects are negligible and high enough that preconditioning is not necessary. A nondimensional time step of $\Delta t = 0.005$ was used to give approximately 500 steps per shedding cycle based on the experimentally-observed Strouhal number of 0.2.

Two variants of the hybrid RANS-LES turbulence model, denoted as HR-DES and HR-LES here, were examined. In HR-DES, as discussed earlier, only the destruction term of is hybridized, resulting in a model very similar to Spalart's DES model. In HR-LES, both production and destruction are hybridized.

The results for this case vary significantly with grid topology. Three different grids were used:

1. Fully unstructured, with tunnel walls, 33 planes over a $0.5c$ span
2. Structured overset, farfield boundaries, 2-D
3. Structured overset, farfield boundaries, 33 planes over a $0.5c$ span

Detailed analysis of the results on these grids is presented in the following sections.



(a) Immediate vicinity of airfoil

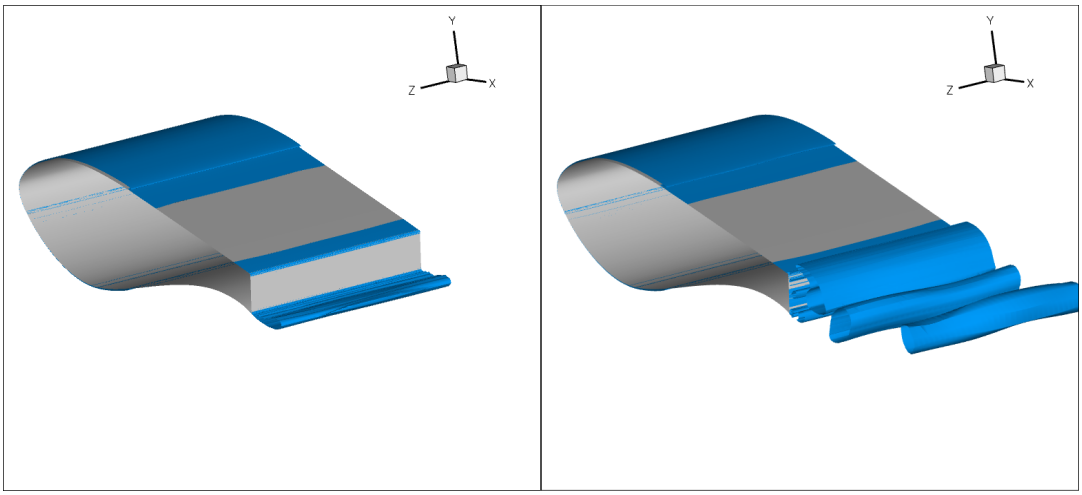
(b) Full grid

Figure 23: Grids used for DU97 simulations with wind tunnel walls. These 2-D grids were extruded in the spanwise direction.

6.1 With Wind Tunnel Walls

Because the wind tunnel configuration has substantial blockage—an airfoil with thickness of about $0.4c$ in a tunnel with a height of $2c$—simulations were first attempted with a grid that included the top and bottom wind tunnel walls. This grid, shown in Figure 23, was generated using Solidmesh [33] and AFLR2 [52, 51] and has a built-in angle of attack of 10°. The grid was generated in two dimensions, and then, to create a 3-D grid, the mesh was duplicated 33 times across a span equal to $0.5c$ or $5h$, where h is the trailing edge thickness. A no-slip boundary condition was used on the airfoil surface, symmetry planes on the side walls, and farfield boundary conditions on the inflow and outflow planes.

Figure 24 shows iso-surfaces of the Q criterion for both k-ω SST and hybrid RANS-LES simulations. It is clear that the k-ω SST results show very little vortex shedding from the blunt trailing edge. The HR-DES results show several vortices in the wake. However, these vortices are essentially 2-D structures. In comparison with the OVERFLOW results of Stone [104], which show clearly 3-D structures, this seems to indicate that the grid here is too



(a) k-ω SST (b) HR-DES

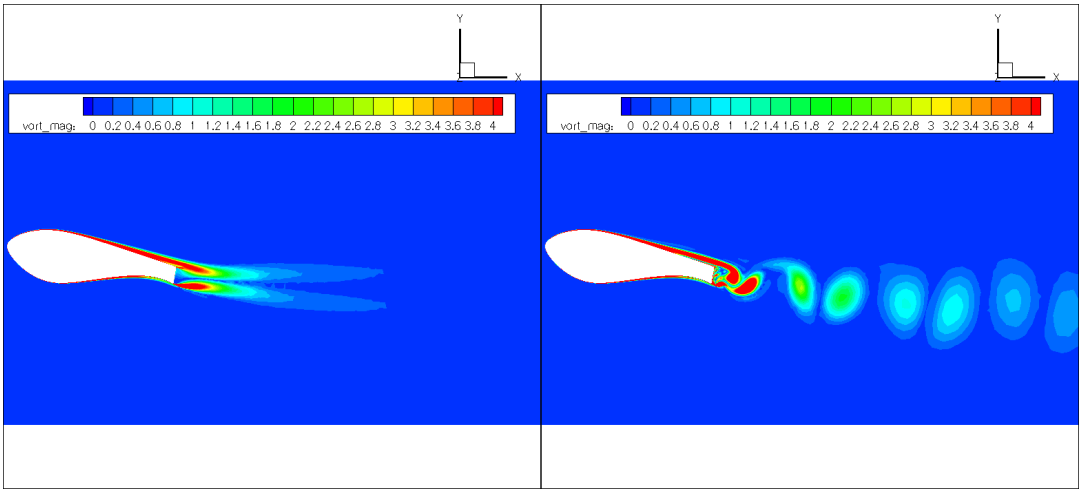
Figure 24: Isosurfaces of $Qc/\hat{a}_\infty = 1.012$ around the DU97 airfoil in the wind tunnel at $\alpha = 10°$

.

coarse in the spanwise direction. A similar conclusion can be drawn from Figure 25 which shows vorticity magnitude at midspan. Nevertheless, the hybrid RANS-LES results show an obviously unsteady flow, with a line of vortices convecting downstream. In contrast, the k-ω SST results, show a pair of stationary vortices just behind the trailing edge, yielding a solution similar to what could be obtained from a steady-state formulation.

Table 2 provides various statistics for the flatback simulations, including mean and RMS lift and drag, as well as Strouhal number. Both k-ω SST and HR-DES yield lift significantly higher than the experimental value. Drag is also higher in the case of HR-DES, which is consistent with the low spanwise resolution. Interpretation of Strouhal number for the two cases is less straightforward. For both, the mode with the highest amplitude occurs at St $= 0.088$. However, the spectra plotted in Figure 26 show that the HR-DES case has a large secondary mode at St $= 0.15$, which is much closer to the experimental value. St $= 0.15$ is also the primary mode in the drag signal.

There are several reasons why the integrated lift is so poorly predicted on grid 1. The primary reason is grid quality. Tangential and normal spacing is very fine at the trailing edge, with 330 nodes across the blunt edge. The comparatively large spanwise spacing ($\approx 0.0156c$) means that cell aspect ratios are quite large at the trailing edge—on the order

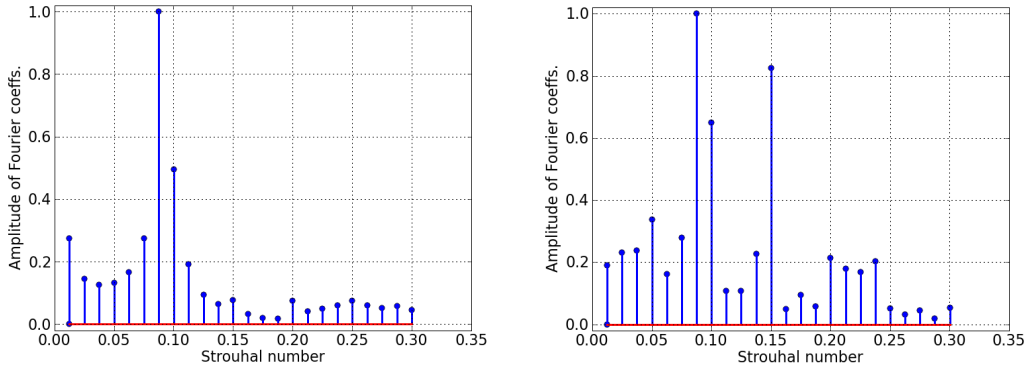


(a) k-ω SST (b) HR-DES

Figure 25: Vorticity magnitude at midspan for the DU97 airfoil in wind tunnel at $\alpha = 10°$.

Table 2: Time-averages and standard deviation of lift and drag for the DU97 flatback airfoil at $\alpha = 10°$. Experimental results were obtained at a nominal $\alpha = 11°$. OVERFLOW results are duplicated from [104]. FUN denotes FUN3D, and OF denotes OVERFLOW.

Grid	Turb. model	Code	$\bar{C}_L$	$\bar{C}_D$	C'_L	C'_D	St
–	Experiment	–	1.57 ± 0.13	0.055 ± 0.005	–	–	0.24 ± 0.01
1	k-ω SST	FUN	1.87	0.0493	0.0160	0.0033	0.088
1	HR-DES	FUN	1.88	0.0740	0.0497	0.0147	see Fig. 26
2	k-ω SST	OF	1.59	0.046	–	–	–
2	HR-DES	OF	1.630	0.103	0.076	0.018	0.213
3	HR-DES	OF	1.56	0.078	0.078	0.017	0.183
3	k-ω SST	FUN	1.615	0.039	0.005	0.001	0.177
3	HR-LES	FUN	1.647	0.061	0.023	0.004	0.182



(a) k-ω SST (b) HR-DES

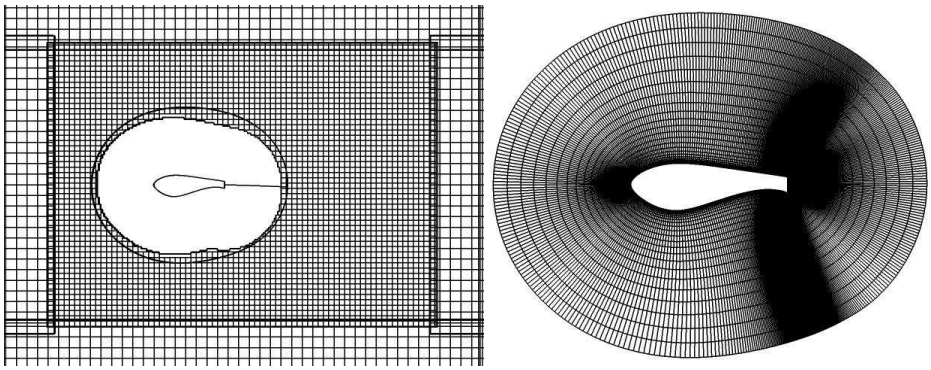
Figure 26: Normalized frequency spectrum of the lift history for the DU97 case. The dual peaks in (b) indicate a secondary shedding mode.

of 100—which was earlier demonstrated to present problems with this hybrid RANS-LES formulation. To make matters worse, the grid cells, which were originally all prisms or hexahedra (due to extrusion), were split into tetrahedra in a pre-processing step. This was done to reduce CPU time since, at the time of these simulations, the pure-tetrahedra path in FUN3D was nearly twice as fast as the more general mixed-element path. Unfortunately, splitting these already fine aspect ratio cells into tetrahedra generated elements with very small angles. Maximum face angles on the dual mesh were over 179°.

In the experimental configuration that grid 1 attempts to emulate, the wind tunnel walls were actually flexible and porous "acoustic windows" made of Kevlar, which allowed sound to reach microphones. That porosity is not accounted for in the boundary conditions used here, nor is the flexing of the walls, which has never been quantified. So the CFD boundaries which were intended to be more realistic than free boundaries may have actually introduced less realistic effects.

6.2 Overset with Free Boundaries

To circumvent problems in specifying correct wall boundary conditions, new simulations were conducted using an overset structured grid without wind tunnel walls. The grid, shown in Figure 27, was originally generated by Stone [104] for use with OVERFLOW. There is a fine O-grid surrounding the airfoil. A series of telescoping Cartesian meshes then extend



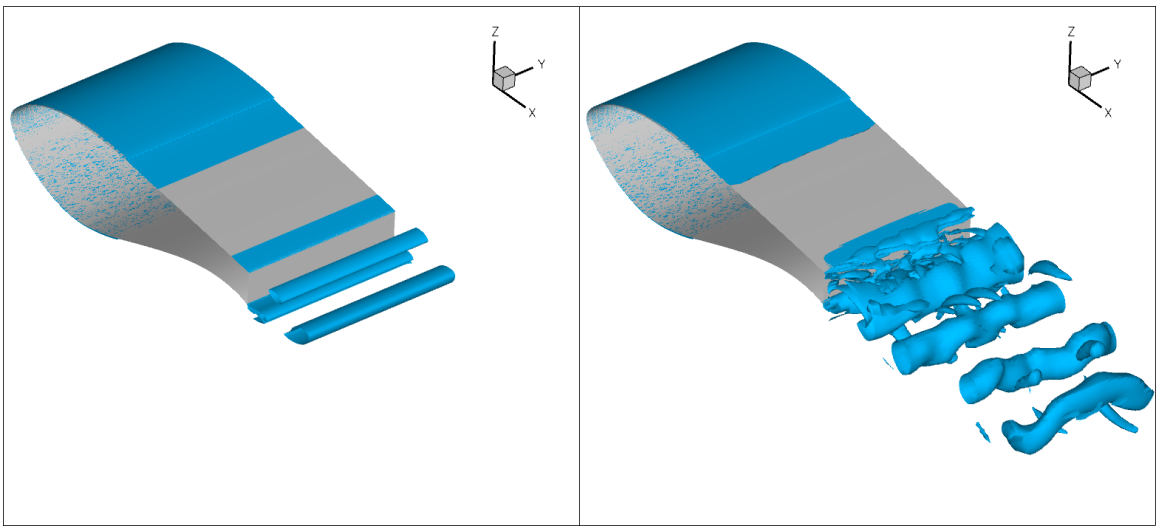
(a) Off-body Cartesian grids (b) Near-body O-grid

Figure 27: Structured overset on DU97 airfoil. Figure and grids taken from Ref. [104].

to the farfield. Each 2-D spanwise plane has 218,650 nodes. To use this grid in FUN3D, the grid was converted to an unstructured hexahedral mesh using SUGGAR++ [64]. Unlike grid 1, the hexahedral cells in this grid were not converted to tetrahedra.

In Table 2, grid 2 corresponds to a 2-D version of this structured overset mesh. Those results, obtained by Stone [104] with OVERFLOW are presented for comparison purposes. Grid 3 is a 3-D version with 33 planes on a $0.5c$ span, for a total of about 7.2 million nodes. On this grid, mean lift is within the experimental range for both codes, on both 2-D and 3-D grids. The drag coefficient with FUN3D on the 3-D grid is also on the edge of the experimental bounds. Finally, on grid 3, the Strouhal number for lift oscillation is somewhat higher than on grid 1, though still not close to that predicted by OVERFLOW on the 2-D grid.

Iso-surfaces of the Q-criterion on grid 3, shown in Figure 28, indicate that HRLES(2) captures a significantly more unsteady wake. The RANS solution is essentially 2-D, while the large rolling vortices shedding from the HRLES(2) solution twist, varying their cross-section as the move dowstream. In addition, near the trailing edge, streamwise braids appear between rollers, a flow feature not visible in HRLES(1) results on grid 1 in Figure 24(b).



(a) k-ω SST (b) HRLES(2)

Figure 28: Q criterion iso-surfaces around the overset DU97 airfoil with free boundaries (grid 3).

CHAPTER VII

HIGH-FIDELITY AEROELASTIC MODELING

The high aspect ratio of modern wind turbine blades makes them very flexible. Due to the simplified aerodynamic models employed, current aeroelastic methodologies do not provide sufficient confidence in their predictions, particularly in loading scenarios near the edges of the operation envelope. As a result, wind turbine structures are usually over-designed, leading to increases in weight and cost. Furthermore, the difficulty in predicting aerodynamic loads on stalled blades is a major factor in the industry trend away from mechanically simple stall-regulated turbines toward more complex pitch-regulated turbines [90].

Though the underlying structural model in typical aeroelastic codes may be quite sophisticated, the aerodynamic models are usually based on some variant of blade element momentum theory [60, 11]. However, advancements over the last two decades in aeroelastic predictions for helicopter applications have demonstrated that using CFD to provide unsteady loads to the structural model dramatically improves loads predictions [77, 26, 3]. Since the aerodynamic methodology has fewer simplifications of the underlying physics, it should have fewer restrictions on the regimes in which it can be applied.

A rigorous validation of a CFD/CSD coupling methodology requires a high quality experimental dataset with which to compare computational results. Unfortunately, at present, there are no such datasets for a wind turbine with non-negligible aeroelastic effects. The NREL Unsteady Aerodynamics Experiment, a very detailed dataset discussed earlier, involved a turbine with relatively short, rigid blades. Therefore, the validation of a CFD/CSD coupling process for a helicopter rotor is presented in this chapter. This process is sufficiently general that few modifications will be required to apply it to a wind turbine.

7.1 HART Rotor

The test case chosen to validate the coupling methodology is the HART-II rotor test [116], which was conducted in 2002 to study the effect of higher harmonic pitch control on rotor

Table 3: Geometric parameters for the HART-II rotor.

Rotor radius, R	2 m
Rotor chord, c	0.121 m
Solidity, σ	0.077
Pre-cone angle	2.5°
Actual shaft tilt, α_{shaft}	5.4° aft
Effective shaft tilt, α_{eff}	4.5° aft
Rotor speed, Ω	108.9 rad/s

noise and vibration. The rotor was situated on a model fuselage, which was at the end of a long sting. Various geometric parameters for the rotor are given in Table 3. To emulate descending flight, the shaft was tilted aft by 5.4 degrees by tilting the entire fuselage/rotor assembly. This has been calculated to correspond to an effective shaft tilt of 4.5 degrees aft when wind tunnel walls are not present. As discussed previously, a number of researchers have demonstrated loose CFD/CSD coupling with the HART-II rotor [46, 50, 16].

7.2 Computational methodology

7.2.1 CSD model

The HART-II rotor lacks flap and lead-lag hinges between the hub and the blades. The blades attach directly to the hub, and the inboard portion of the blade, which has an elliptical cross-section and is much stiffer than the outboard portions, is allowed to bend elastically to absorb some of the bending moment that would otherwise be transferred to the hub. The DYMORE model used in these simulations, shown in Figure 29, has a number of simplifications compared to the actual rotor. First, most of the hub hardware is omitted. The blades simply attach to a revolute joint at the hub axis. While the HART blade is physically a single piece, here each blade is modeled as two beams: a relatively stiff inboard "flex beam" and a more flexible outboard main blade beam. The flex beam is constructed from a single third-order finite element. The main blade consists of eight third-order elements. Figure 30 shows the relationship between the number of main blade elements and the first torsion frequency. With eight finite elements, the first torsion frequency can be computed to within 0.5% of the measured frequency. It is important to minimize the number of elements used in the model since DYMORE's performance scales with $O(N_e m^2)$,

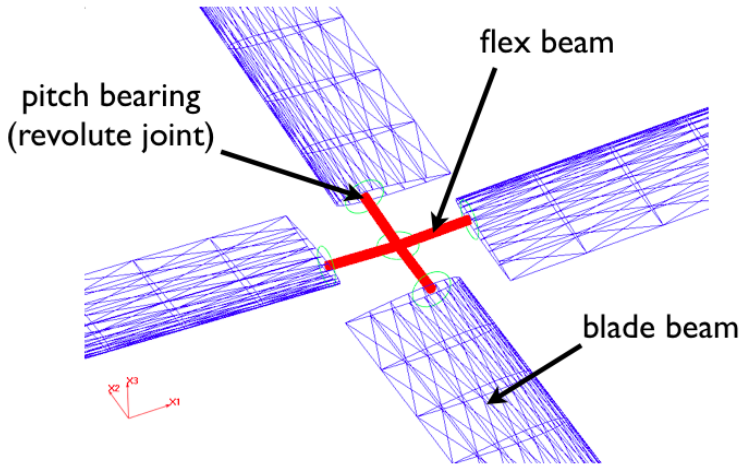


Figure 29: Hub of the HART DYMORE model.

where N_e is the number of elements, and m is the bandwidth of the stiffness matrix [9]. Since adding more one-dimensional beam elements does not change m, performance scales linearly with N_e.

The pitch link is also omitted since there is a great deal of uncertainty as to its structural properties. To approximate the effect of the control system stiffness on the blade response, a torsional spring connects the flex beam and the main blade beam. In order for the torsional spring to provide the best possible approximation of the control system stiffness, the spring constant must be adjusted until the natural frequency associated with the first torsional mode matches measured values. This "tuning" process resulted in a stiffness of 1632.4 Nm/rad, yielding the target natural frequency of 419.57 rad/s. Figures 31 and 32 show a fan plot and the first six mode shapes at the nominal frequency, respectively. The computational model follows the trend of the measured frequencies, and the mode shapes are nearly identical to the measured modes.

As stated in Table 3, the HART blades had a small pre-cone angle of 2.5°, which was included in the DYMORE model in the reference configuration. This means that the deflections output by DYMORE are relative to the pre-coned state. The aft shaft tilt was applied to the model via the farfield velocity used by the aerodynamic interface, which mirrors the manner in which shaft tilt was applied to the CFD model by setting the angle of attack to 4.5°.

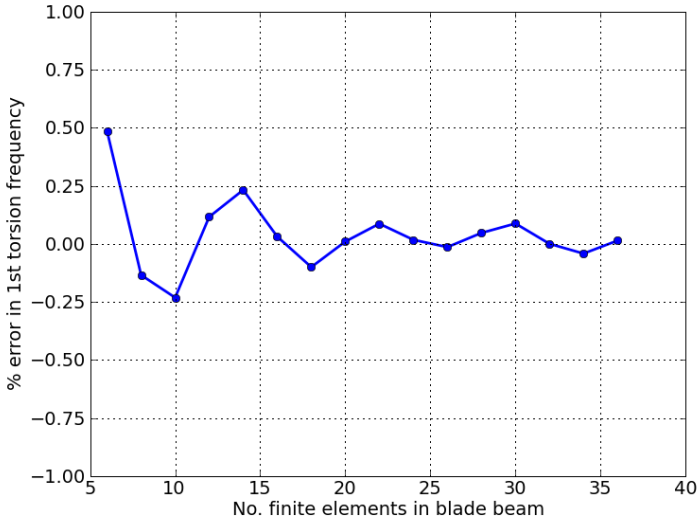


Figure 30: Convergence of the 1st torsion frequency with the number of finite elements in the blade beam.

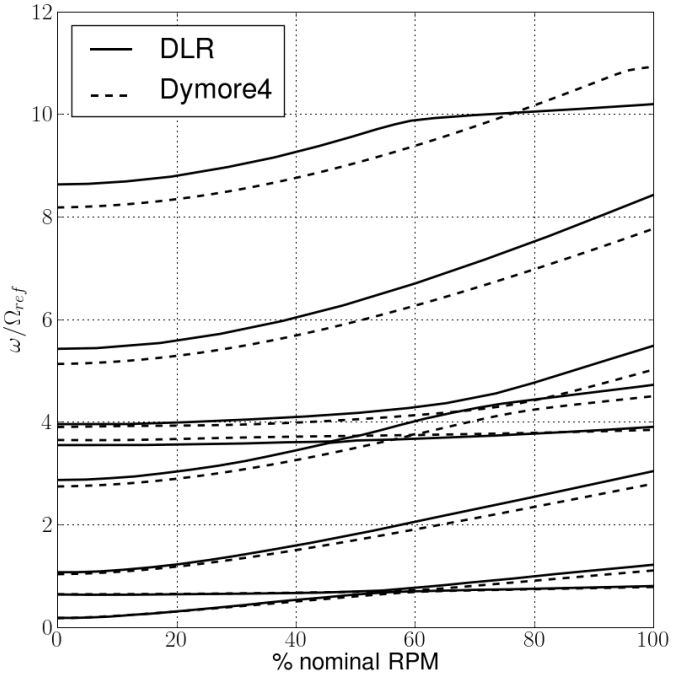


Figure 31: Fan plot for the HART DYMORE model.

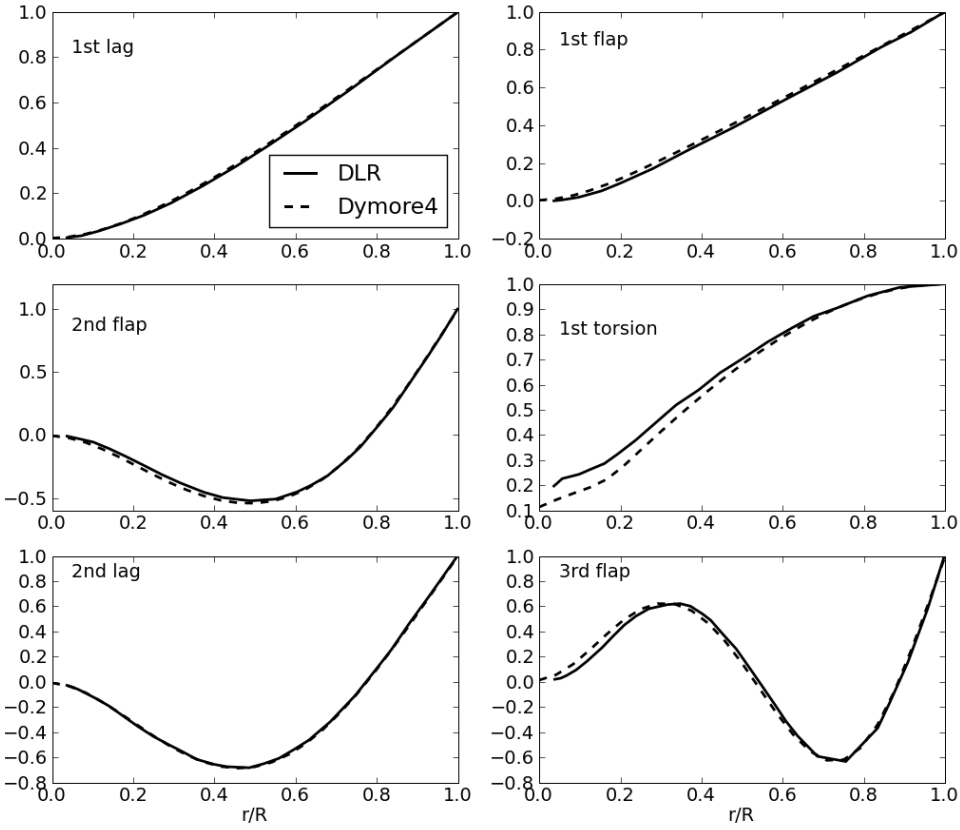


Figure 32: First six mode shapes for the HART Dymore model.

Table 4: Dimensions of each component of the composite mesh. All cells are tetrahedra.

Component	Nodes	Boundary faces	Cells	Grid generator
Blades (each of 4)	1,156,735	114,302	6,731,961	VGRID
Fuselage/background	8,971,420	63,026	52,733,053	VGRID
Total	13,598,360	520,234	79,660,897	

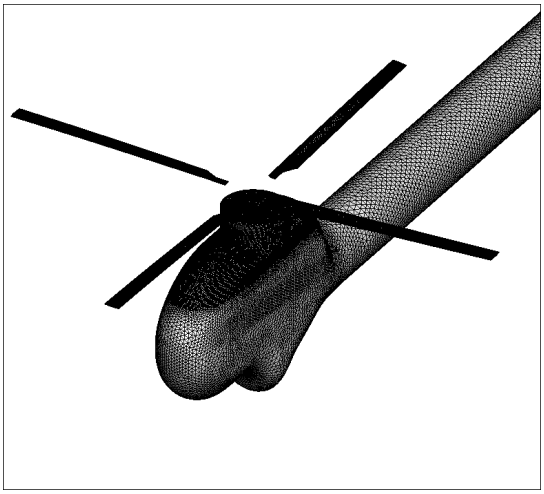
7.2.2 CFD model

The mesh used for these simulations (Figure 33) included a total of 13.6 million nodes. The relevant mesh statistics are summarized in Table 4. This mesh is identical to that used by Biedron and Lee-Rausch in 2008, and as described in Reference [16], this mesh was the finest of a set of meshes used in a grid convergence study.

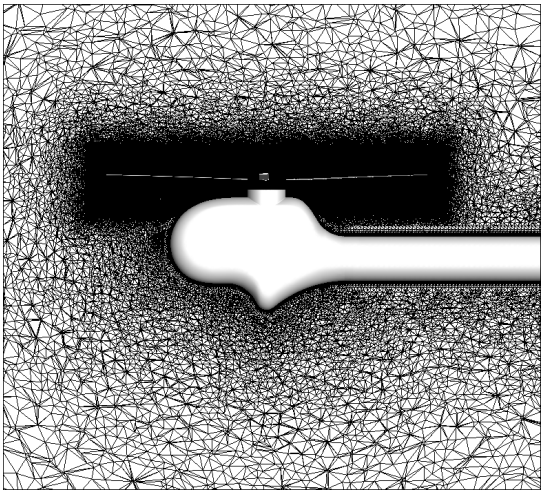
The CFD portion of the coupled simulation followed as closely as possible the work of Biedron and Lee-Rausch [16], in order to provide a correlation of the loosely coupled analysis prior to tight coupling. A timestep corresponding to one degree of azimuthal rotation was used, and the number of Newton subiterations was dynamically determined by a temporal error controller [111], insuring that at each time step, residuals were always reduced to less than 5% of an estimate of the temporal error. On average, using the error controller resulted in about 30 Newton subiterations per step. To correlate with Reference [16], the Spalart-Allmaras turbulence model was selected.

7.2.3 Coupling procedure

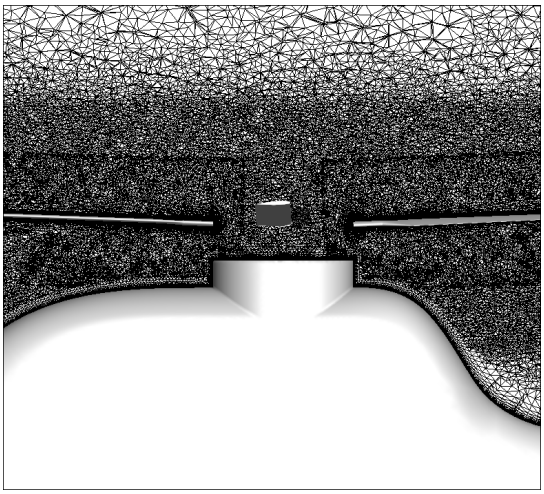
The goal of this work is to demonstrate a tight coupling method. However, this presents some difficulties when trimming the rotor. During loose coupling, the CSD solver is provided loads at all the airstations for a single revolution of the rotor. It then enters a "reference" phase in which those loads are applied to the CSD model until the motion has become periodic, usually after about 50 revolutions. Then the CSD trimmer enters a "perturbation" phase in which each of the three pitch controls is perturbed by a small amount one at a time. The perturbation of each control lasts for 5-10 revolutions. After perturbing all three controls, the change in hub loads resulting from each perturbation is used to construct an inverse Jacobian matrix. Then, that inverse Jacobian is used to compute a new set of controls, and the CSD solver enters the "simulation" phase. Finally, it is allowed to



(a) Surface grid



(b) Volume grid (sliced through centerline)



(c) Volume grid (close-up of hub region)

Figure 33: 13.6 million node composite mesh used for HART-II simulations.

run until the loads again become periodic, about 100 revolutions. This procedure is often termed an auto-pilot trimmer. Though over 150 revolutions may have been simulated in total, the CSD solver is typically quite efficient and requires no further inputs from the CFD solver, so this is not an unacceptable computational burden. For example, one auto-pilot trim cycle in DYMORE usually requires about one hour, depending on various factors, such as the number of airstations.

In tight coupling, the same trim procedure described above would be *extremely* expensive since the CFD solver would have perform as many revolutions as the CSD solver. A number of alternative trim algorithms have been proposed to deal with this situation, but those are still an area of ongoing research [37]. For this effort, trim is computed with a conventional loose delta coupling method until convergence has been reached, after which the simulation is switched to a tight coupling algorithm. The initial loose coupling procedure (for trimming) follows that used by Abras [3] and Biedron and Lee-Rausch [16]. The only difference is that a quasi-steady trimmer was used in place of the auto-pilot. With a quasi-steady trimmer, the user supplies an inverse Jacobian matrix (perhaps computed in a single auto-pilot run), and that matrix is used to continually adjust the controls to drive the mean hub loads to their target values.

After switching to tight coupling, the controls are kept fixed. Provided the initial loose coupling converged, a correctly functioning tight coupling method should result in minimal change in the solution during its operation even though the elastic deformations can change.

7.3 Results

7.3.1 Validation of the coupled model

The loose coupling phase was allowed to converge via five coupling iterations. The "zeroth" iteration was run for two revolutions, and subsequent iterations each ran for half a revolution. It should be noted that for a rotor with n_b blades, a coupling iteration need only last $1/n_b$ revolutions. At that point, the airloads from all the blades can be combined to create a map of airloads over an entire revolution. However, as the blade controls undergo a step change at the beginning of each coupling iteration leading to transients in airloads, running

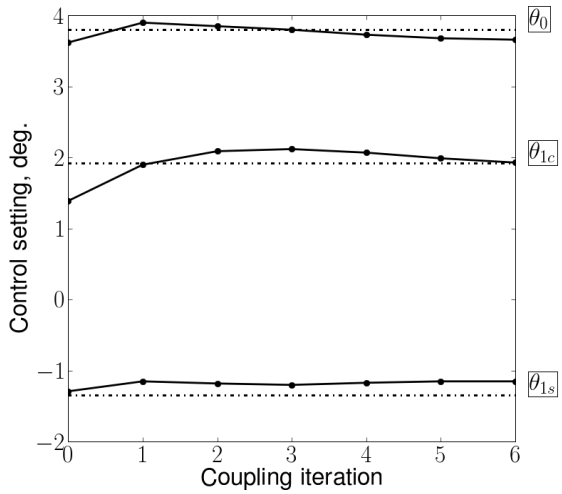


Figure 34: Convergence of blade pitch controls with loose coupling iterations.

Table 5: Converged blade pitch controls, in degrees.

	θ_0	θ_{1c}	θ_{1s}
Measured	3.80	1.92	-1.34
Computed	3.66	1.93	-1.15
Computed (Ref. [16])	3.47	1.77	-0.97

$2/n_b$ revolutions allows the airloads to become quasi-periodic state before applying them to the DYMORE model.

Figure 34 shows the convergence of blade collective and cyclic pitch, θ_0, θ_{1c}, and θ_{1s}, during the initial loosely coupled trim phase. The controls change very little after the second iteration, though as will be shown later, the airloads continue to change. Only the DYMORE portion of the sixth iteration was simulated. The controls from the sixth iteration were then fixed for the later tight coupling phase. The final controls are given in Table 5, along with controls computed via FUN3D/CAMRAD coupling from Reference [16].

The convergence of airloads at 87% span is shown in Figure 35. Prior work with the HART rotor shows that many predictions, including airloads and structural loads, have finite offsets from the measured values. Since the fluctuating component is of greater engineering interest than the mean value, Figure 35(c) shows pitching moment with the mean component removed to show the fluctuating component only. It is clear that by the fifth loose coupling iteration, the airloads have nearly converged on their final values. The primary

Table 6: Mean values subtracted from structural moments in second column of Figure 36.

	Flap, Nm $r/R = 0.17$	Torsion, Nm $r/R = 0.33$	Lag, Nm $r/R = 0.17$
BL measured	-9.27	-2.90	-10.26
FUN3D/DYMORE	-3.05	-5.17	-64.49
FUN3D/CAMRAD	-3.36	-5.18	-63.18

differences between the ultimate and penultimate iterations are observed for the normal force predictions in the first and fourth quadrants, which feature prominent blade-vortex interactions. Since the controls remain relatively constant after the second iteration, it can be concluded that most of the adjustments made by DYMORE after that point are elastic deformations in an attempt to resolve the BVI phenomena.

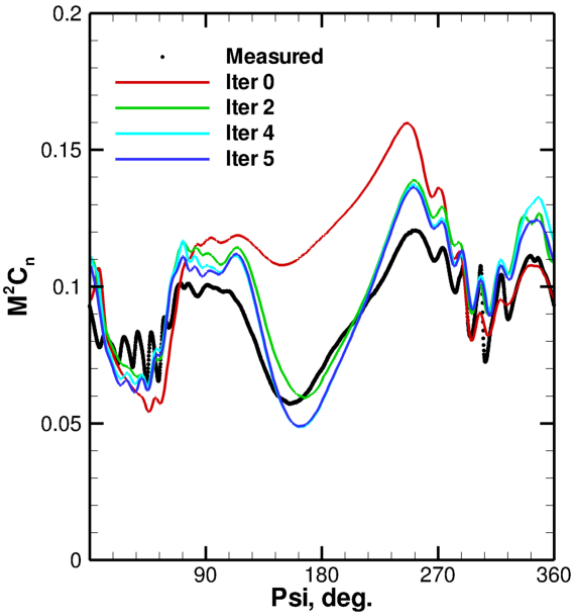
In an effort to further establish the validity of the structural model, structural moments from the present simulation were compared against measured moments and those computed by Biedron and Lee-Rausch [16]. The present loosely coupled simulations capture the character and magnitude of the flap moment better than FUN3D/CAMRAD. The torsion and lag moments in Figure 36 are nearly identical, and DYMORE has the same difficulty as FUN3D/CAMRAD in capturing the lag moment, which is very lightly damped. For reasons discussed earlier, the second column of Figure 36 shows the structural loads with their means removed. Those mean values are listed in Table 6.

A comparison of FUN3D/DYMORE airloads from the final loose coupling iteration with those predicted by FUN3D/CAMRAD is provided in Figure 37. The two coupling methodologies provide similar results at all azimuths, with FUN3D/DYMORE capturing the BVI regions better, particularly the one in the fourth quadrant.

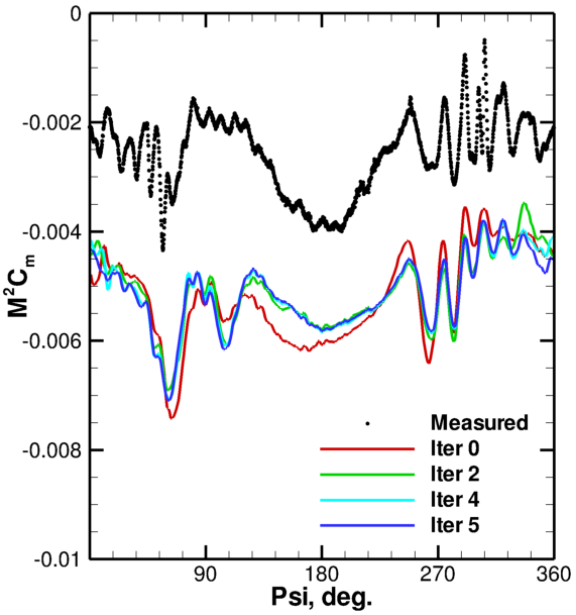
7.3.2 Evaluation of tight coupling

When a tightly coupled solution is initialized from a previously converged loosely coupled solution, the computed elastic deformations and airloads should not change.

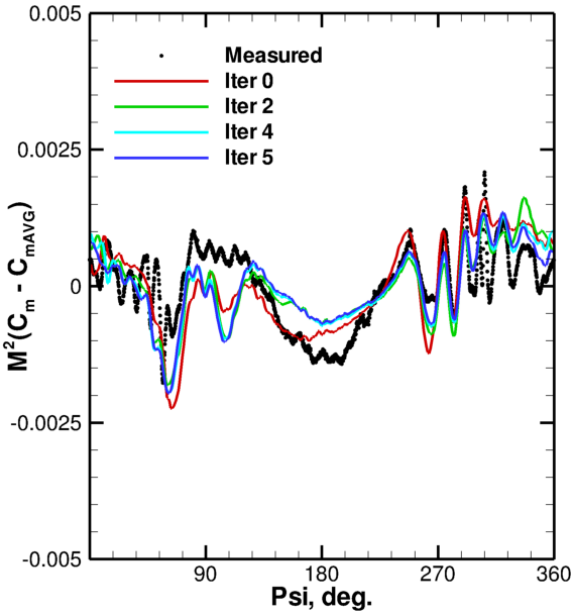
Tip torsion, flap, and lag predicted after switching to tight coupling are shown in Figure 38, along with the motions from the final iteration of loose coupling. There is very little change in torsion and flap, although there is a small decrease in flap near 180° azimuth.



(a) Normal force, $C_n M^2$

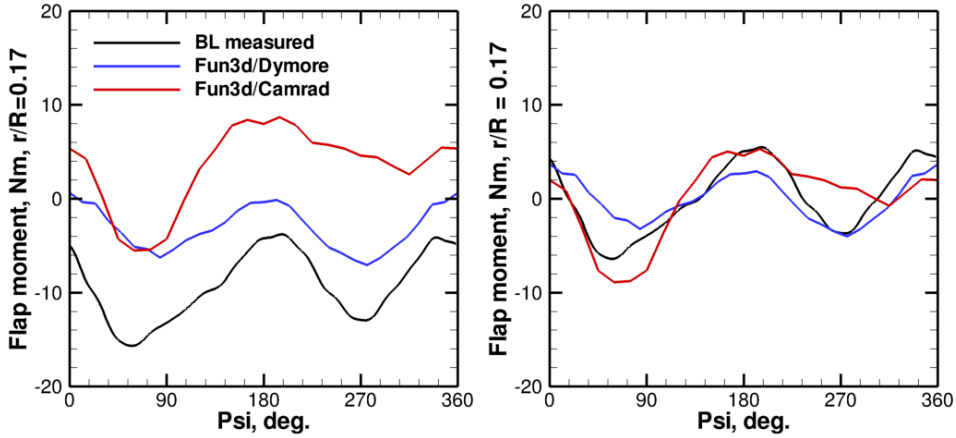


(b) Pitching moment, $C_m M^2$

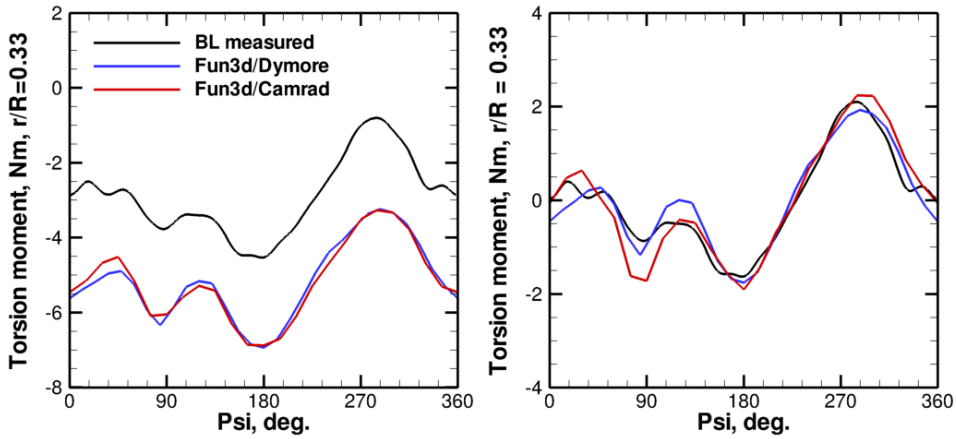


(c) Pitching moment with mean removed

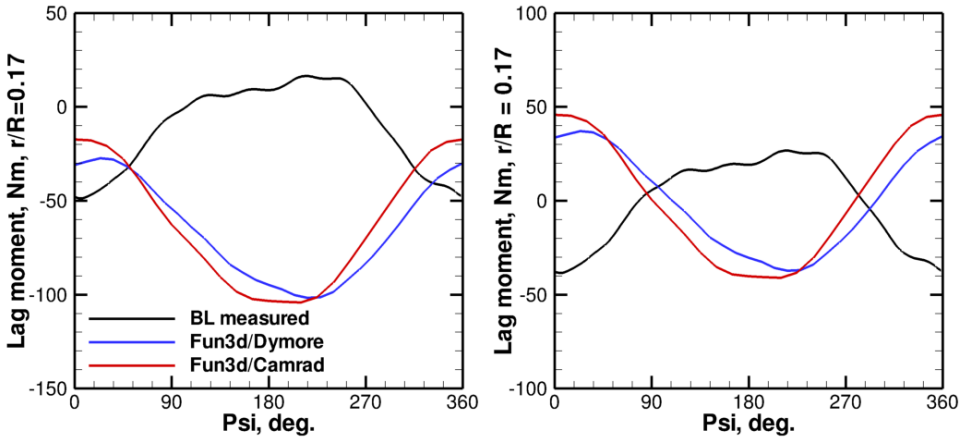
Figure 35: Convergence of airloads at 87% span for the loosely coupled trim phase. For measured pitching moment, the mean value is -0.00258; for iteration 5, the mean pitching moment is -0.00513.



(a) Flap moment at $r/R = 0.17$ (b) Flap moment, mean removed

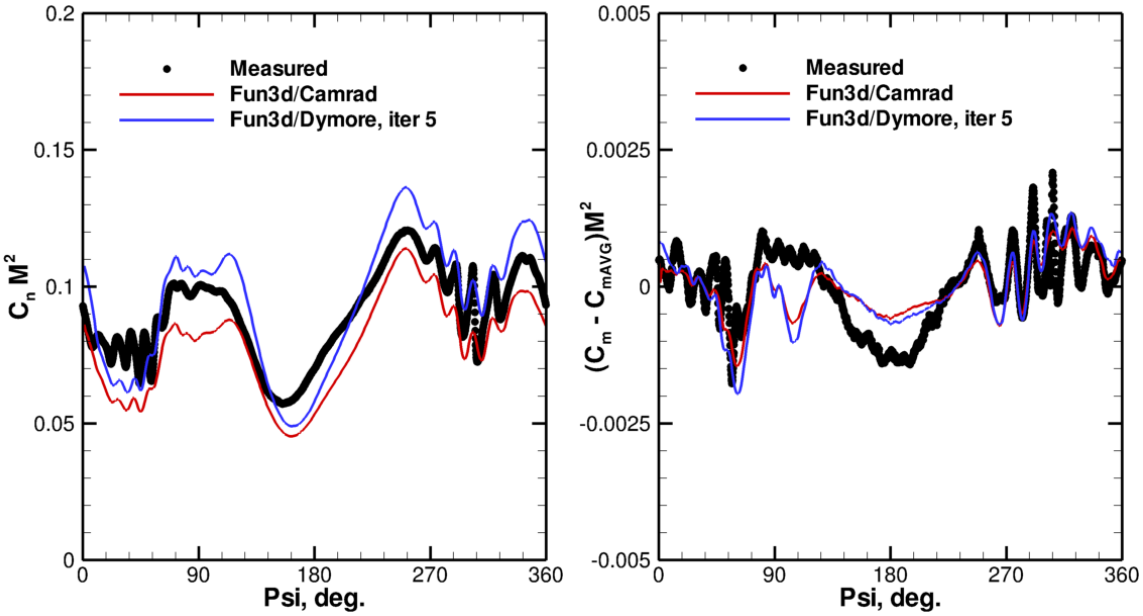


(c) Torsion moment at $r/R = 0.33$ (d) Torsion moment, mean removed



(e) Lag moment at $r/R = 0.17$ (f) Lag moment, mean removed

Figure 36: Comparison of structural moments computed using FUN3D/DYMORE and FUN3D/CAMRAD in Reference [16]. Mean values removed in second column are listed in Table 6.

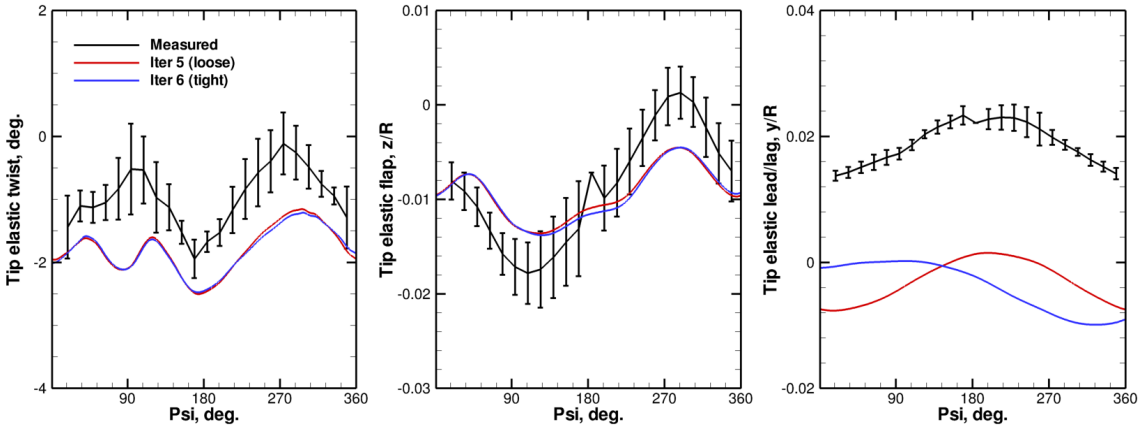


(a) Normal force, $C_n M^2$ (b) Pitching moment, $C_m M^2$, with mean removed

Figure 37: Comparison of airloads at 87% span computed using FUN3D/DYMORE and FUN3D/CAMRAD in Reference [16]. Mean pitching moments are -0.00258, -0.00424, and -0.00513 for measured, FUN3D/CAMRAD, and FUN3D/DYMORE, respectively.

Airloads at 87% span, shown in Figure 39, also show minimal change, particularly in pitching moment. Due to the small change in flapping motion at 180°, there is a small change in normal force. There are other small variations in normal force in the first quadrant BVI events. Since the tightly coupled solution moves closer to the measured data, it can be surmised that the initial loosely coupled solution was not completely converged. Otherwise, the solution is quite stable.

The mean measured thrust is $T = 3300$ N. The mean measured rolling and pitching moments are $M_x = 20$ N-m (positive for rolling to the left) and $M_y = -20$ N-m (positive for nose-up pitching). At the end of loose iteration five, the computed mean hub loads were $T = 3326$ N, $M_x = 11.9$ N-m, and $M_y = -29.2$ N-m. After a revolution of tight coupling, the hub loads are nearly the same, with $T = 3328$, $M_x = 10.2$ N-m, and $M_y = -10.7$ N-m. Though the change in pitching moment as a fraction of the target value is large, instantaneous pitching moment varies from -200 N-m to 160 N-m, so the change in the mean is only 5.1% of the total variation.



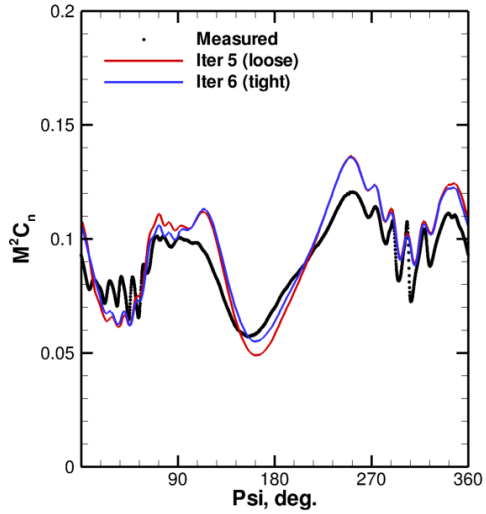
(a) Elastic torsion (b) Flap (c) Lead-lag

Figure 38: Tip deflections predicted in tight coupling and the final iteration of loose coupling. Bars on the measured data indicate blade-to-blade variation.

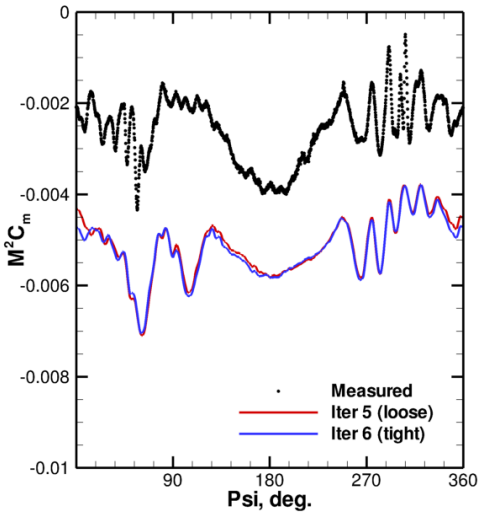
Iso-surfaces of the second invariant of the velocity gradient tensor, also known as the "Q criterion", are shown in Figure 40. Q criterion, defined earlier in Equation 23, has the property that it is only positive near vortices. As such, it is a useful tool for distinguishing between sheets of high vorticity and actual vortices. The roll-up of the individual tip vortices into a single horseshoe vortex is clearly visible in these visualization, though the horseshoe dissipates just after of the rotor disc after exiting the refined region of the background grid. Just before 90° and just after 270° are regions where the blade tip can be observed to pass through several tip vortices in rapid succession, leading to the blade-vortex interactions discussed previously. In Figure 40(a), it is also clear that the system of tip vortices resides primarily within the plane of the rotor, as one would expect since the rotor is tilted aft to simulate a descending flight case.

7.4 Performance

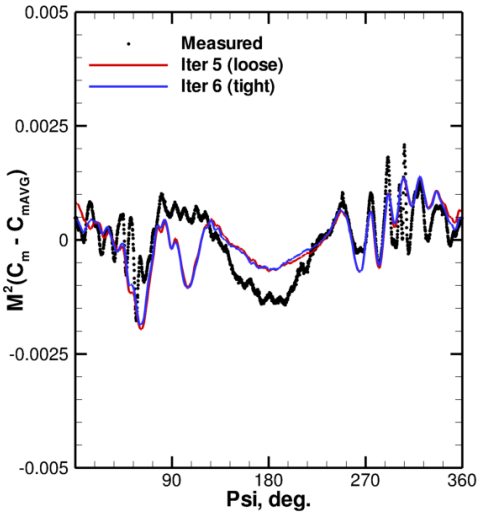
The flow solution was run on a Cray XT5 supercomputer. Each cluster node had two 2.3 GHz quad-core CPUs, for a total of eight cores per node. Four hundred cores were used for the flow solution. A single core was used for hole-cutting with the other seven cores on that node sitting idle to increase the memory available to the hole-cutting process. Each revolution required 7.3 hours of wall time. A total of 5.5 revolutions were simulated,



(a) Normal force, $C_n M^2$

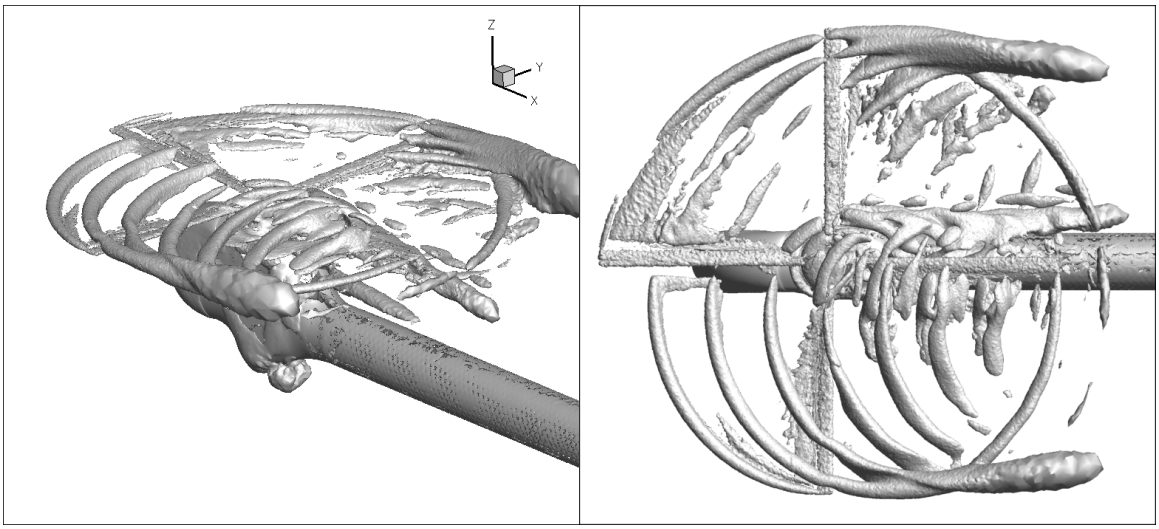


(b) Pitching moment, $C_m M^2$



(c) Pitching moment with mean removed, $(C_m - C_{mAVG})M^2$

Figure 39: Sectional normal force and pitching moment 87% span, comparing loose and tight coupling.



(a) (b)

Figure 40: $Q = 0.0075$ iso-surfaces. Q is non-dimensionalized by speed of sound and reference length, 1 m.

including the final revolution of tight coupling, resulting in a total wall time of about 84 hours or 34,272 CPU hours. During tight coupling, DYMORE had a negligible impact on CPU time and memory usage (less than 1% each). DYMORE required about 45 minutes to trim each loose coupling iteration on a single processor. The SUGGAR hole-cutting and overset assembly process, which runs as a single process on the first MPI rank, takes 25% of the total wall clock time in both loose and tight coupling. A single overset assembly process is currently used because the mesh partitioning scheme in FUN3D, which yields compact partitions that minimize communication (and therefore maximize scalability in the flow solver), limits the parallel scalability of SUGGAR [64].

CHAPTER VIII

OVERSET WIND TURBINE SIMULATIONS

Currently, the most sophisticated technique to accurately model rotating systems, such as a wind turbine, is an *overset* CFD simulation. Such a simulation can capture the unsteady physics associated with the movement of the blades as well as the effect the tower and nacelle have on the vortical wake. This chapter presents results from simulations of an upwind, two-bladed turbine with the goal of predicting rotor performance as accurately as possible.

8.1 NREL Unsteady Aerodynamics Experiment

The NREL Phase VI Unsteady Aerodynamics Experiment, completed in 2000 [35], was an extensive series of tests of a two-bladed, stall-regulated wind turbine in the 80×120' wind tunnel at the NASA Ames Full Scale Aerodynamic Complex. The 5-m blades were instrumented with a large number of pressure taps and strain gauges to record unsteady blade loads and structural responses. Numerous operating conditions were tested, including both upwind and downwind rotor configurations at wind speeds ranging from 5 to 25 m/s and yaw angles from 0 to 30 degrees. The purpose of the tests was to build a dataset for use in studying the complex aerodynamic interactions of a wind turbine and for improving computational prediction of structural loads. Even though modern turbines are always pitch-regulated, the Phase VI rotor has become a *de facto* standard for evaluating aerodynamic prediction methodologies.

Following completion of the wind tunnel tests, NREL initiated a "blind" comparison [90], in which researchers were invited to use their codes to predict the turbine performance at each operating conditions. Many different types of prediction methodologies were used by the participants, including blade element methods, free wake methods, and three different CFD methods. Though the Phase VI turbine is relatively small by the standards of a modern utility-scale turbine, it exhibited many of the same aerodynamic phenomena as

Figure 41: NREL Phase VI turbine mounted in 80 × 120' tunnel.

larger machines. These include 3-D, unsteady flow over the blades, stall at higher wind speeds, and stall-delay due to rotational augmentation [84, 82, 83]. In the blind study, there was very little agreement in the different performance and loads predictions, even in low-speed, unstalled cases.

In the present work, selected cases in the upwind baseline configuration (Sequence S), were simulated using an unstructured overset methodology. The nominal blade tip pitch in this case was 3 degrees. The leading edge pressure probes normally present on the blades were removed and the flow was allowed to freely transition to turbulence (as opposed to being tripped). This sequence of cases was chosen in part to allow correlation with the unstructured CFD simulations of Potsdam and Mavriplis [74]. Comparison of sequence S with sequence M, in which the boundary layers were tripped near the leading edge, shows that at the wind speeds simulated here, transition effects are minimal, which is important since the computational methodology assumes fully turbulent flow. As will be discussed later, however, the issue of transition is not quite as straightforward as whether the boundary layer is tripped.

8.2 Description of mesh and computational method

In an overset simulation, the geometry is broken into smaller components, each of which is meshed independently of one another, and then reassembled and permitted to overlap

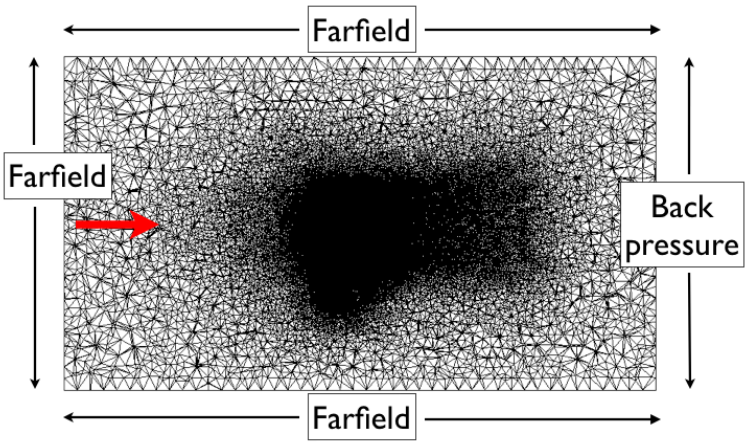
Table 7: Dimensions of each component of the composite NREL Phase VI mesh. Boundary layer cells are triangular prisms; non-boundary layer cells are tetrahedra.

Component	Nodes	Boundary faces	Cells	Grid generator
Blades (each)	4,510,177	179,570	17,687,963	VGRID surface mesh, AFLR3 volume mesh
Nacelle and tower	971,059	51,620	2,286,757	VGRID
Background	4,776,082	10,488	28,278,639	AFLR3
Total	14,767,495	421,248	67,084,524	

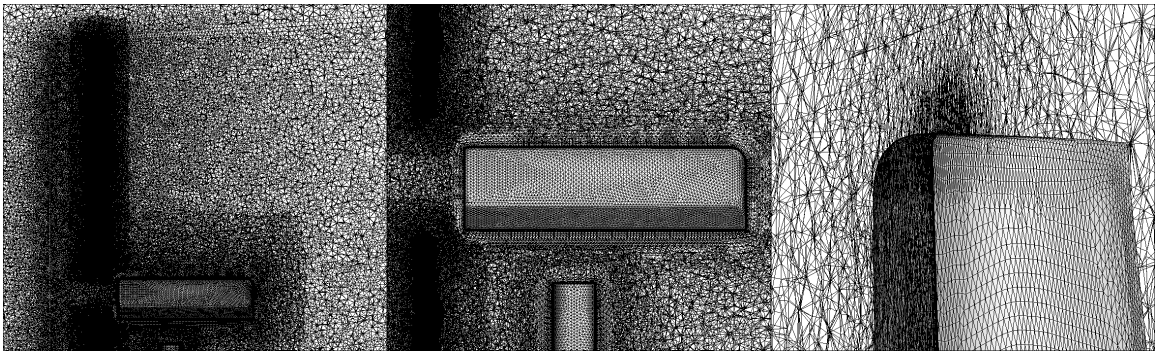
arbitrarily. Any node of one mesh that lies inside solid boundaries defined by another mesh must be marked in such a way that it does not contribute to the solution. This marking process is called "hole-cutting." In overlap regions away from boundaries, an interpolation or mapping procedure must be used to communicate flow information from one mesh to those it overlaps.

If the meshes are structured, overset methods simplify the grid generation process since component meshes can be chosen so that it is easy to "wrap" a body-fitted mesh around each component. Unstructured methodologies do not suffer from the difficulty of fitting a mesh to a complex geometry, so unstructured overset techniques merely facilitate relative movement between the components.

The composite mesh for the Phase VI rotor consisted of four component meshes: two blades, the nacelle and tower, and a background mesh. Various statistics for the component meshes are given in Table 7. The final composite mesh, shown in Figure 42, had over 14.7 million nodes. The blade grids were generated using a combination of VGRID [73] and AFLR3 [52]. The former was used to provide a surface grid that was anisotropically stretched in the spanwise direction. This dramatically reduces the overall node count of the mesh and is a capability unique to VGRID. AFLR3 was used to build a mixed-element volume grid with a boundary layer consisting primarily of prismatic elements. The initial normal spacing on the blade grids was iteratively refined until $y+ < 1$ over the entire blade. Background grid spacing in the vicinity of the rotor was 0.03 m, which is just under 10% tip chord, as recommended by Potsdam et al. [75].



(a) Whole domain



(b) Rotor region

(c) Nacelle

(d) Blade tip cutting through background

Figure 42: The final composite mesh for the NREL Phase VI turbine.

Table 8: Integrated turbine loads. Asterisks in the last column indicate cases that were impossible to trim. The given pitch is the value that resulted in thrust closest to the target value.

Wind speed (m/s)	Yaw (deg)	Turbulence model	Thrust (N)	Torque (N-m)	Root flap bending (N-m)	Trimmed pitch (deg)
10	0	Meas.	1664	1313	2471	3.0
		k-ω SST	1821	1808	3055	5.0*
		HR-LES	1821	1808	3055	5.0*
15	0	Meas.	2278	1104	3040	3.0
		k-ω SST	2660	1670	4165	1.0*
		HR-LES	2301	1372	3496	4.0
15	30	Meas.	2112	1483	2989	3.0
		HR-LES	2187	1591	3498	2.5

Hole-cutting and grid assembly were accomplished using SUGGAR++ [64]. Donor interpolation within FUN3D was handled by DIRTLIB [62]. Due to grid motion, new holes had to be cut at each time step. This was facilitated by FUN3D's "DCI-on-the-fly" capability.

8.3 Results

Zero-yaw cases were run at wind speeds of 10 and 15 m/s, using the k-ω SST and HR-LES (full hybridization) turbulence models for each case. In each case, the rotor was "trimmed" by manually adjusting the blade pitch until the predicted thrust matched the measured thrust. This provides a means to correct for any errors in measurement of the blade pitch in the wind tunnel tests as well as any small differences between the published blade twist and the twist in the mesh. Thrust, defined here as force directed along the rotor axis that is averaged over a revolution, is not normally a quantity of interest for wind turbines, but is less difficult to predict than torque [59]. The nominal tip pitch of 3° was used as the starting point for the trim process. A trimmed solution was not possible in every case. The manual trimming process requires that thrust be monotonic with pitch, which is not necessarily the case when the rotor is stalled. In addition, any trim solution that deviated more than two degrees from the nominal pitch was rejected. The mean loads for each case, as well as the trimmed pitch, are listed in Table 8.

8.3.1 10 m/s, 0 degrees yaw

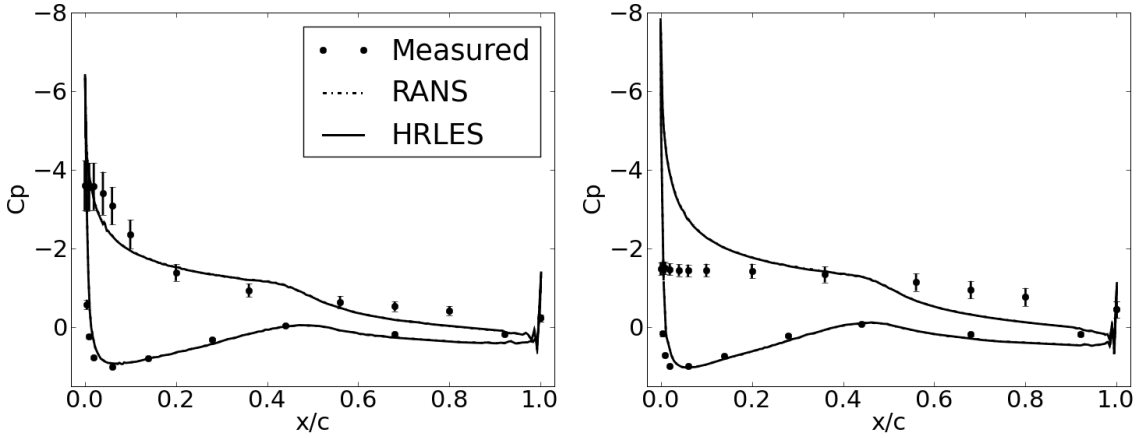
Pressure coefficients at 10 m/s and 0° yaw are shown in Figure 43. In the measured pressure, it is clear that this case has little unsteady flow aside from an area near the leading edge at $r/R = 0.3$, as denoted by the bars that indicate the extent of the unsteady measurements. As a result, the RANS and HR-LES solutions are nearly identical, with no observable difference in mean pressure. At the three outboard stations, C_p predictions are quite close to the measured data. At $r/R = 0.3$, the overall trend is captured, but the leading edge shedding is not captured, resulting in a lower C_p prediction. The flat C_p profile over the first half of the chord at $r/R = 0.47$ is not predicted by either RANS or HR-LES. Due to the large over-prediction of suction C_p at $r/R = 0.47$, where the force integration panel area is equal to 25% of the total planform area, neither RANS nor HR-LES was able to reach a physically meaningful trimmed solution. A tip pitch of 5° was reached before trimming was abandoned.

The flat C_p profile at $r/R = 0.47$ indicates separation, and the fact that the separation occurs at the leading edge suggests that it is a laminar separation, followed by reattachment. However, as stated earlier, transition seems to be of minimal importance in these cases. A more likely cause of the poor prediction is the low level of freestream turbulence in the present simulations. The turbulence intensity in the NASA/Ames wind tunnel is roughly $\sqrt{2k/3}/U = 0.005$ [97], but a value of 7.7×10^{-5} (the FUN3D default) was used here.

Wake visualizations for the 10 m/s case with HR-LES are shown in Figure 44. Tip vortices are resolved for 200 degrees of wake age. The disruption of the tip vortices downstream of the tower is visible in the vortex just behind the tower. The C_p profiles in Figure 43 show fully attached flow in the computational results (suction surface pressure increasing monotonically from leading to trailing edge), which agrees with the wake visualizations, which exhibit no spanwise vortex shedding.

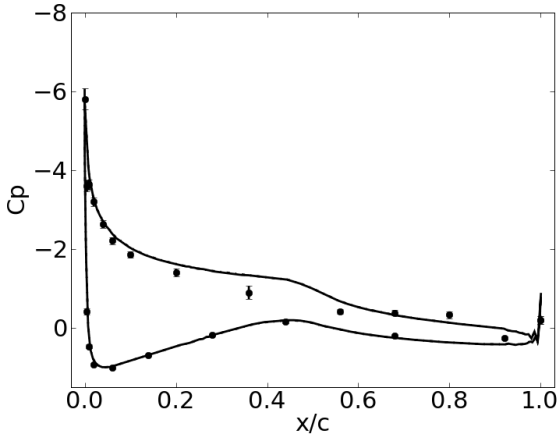
8.3.2 15 m/s, 0 degrees yaw

Mean pressure coefficients at 15 m/s in Figure 45 show distinct differences between RANS and HR-LES solutions. At this wind speed, the measured data show that the blade is

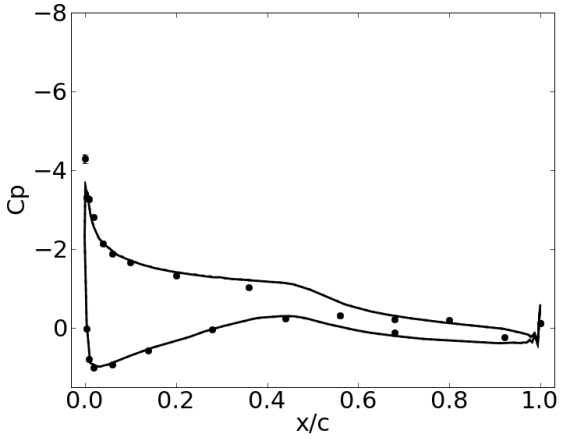


(a) $r/R = 0.30$

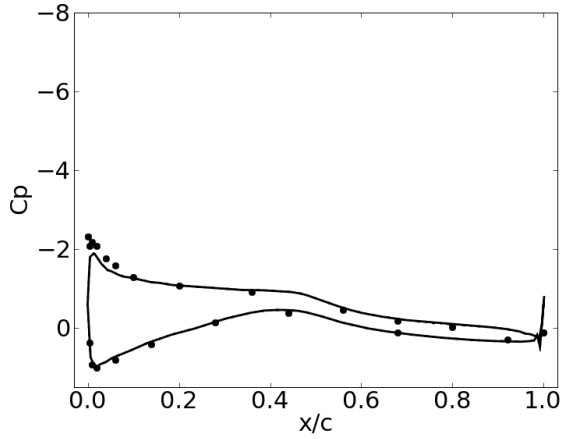
(b) $r/R = 0.47$



(c) $r/R = 0.63$



(d) $r/R = 0.80$



(e) $r/R = 0.95$

Figure 43: Mean sectional pressure coefficient at 10 m/s. Bars indicate standard deviation of pressure at a particular tap, not error. RANS and HR-LES results are coincident.

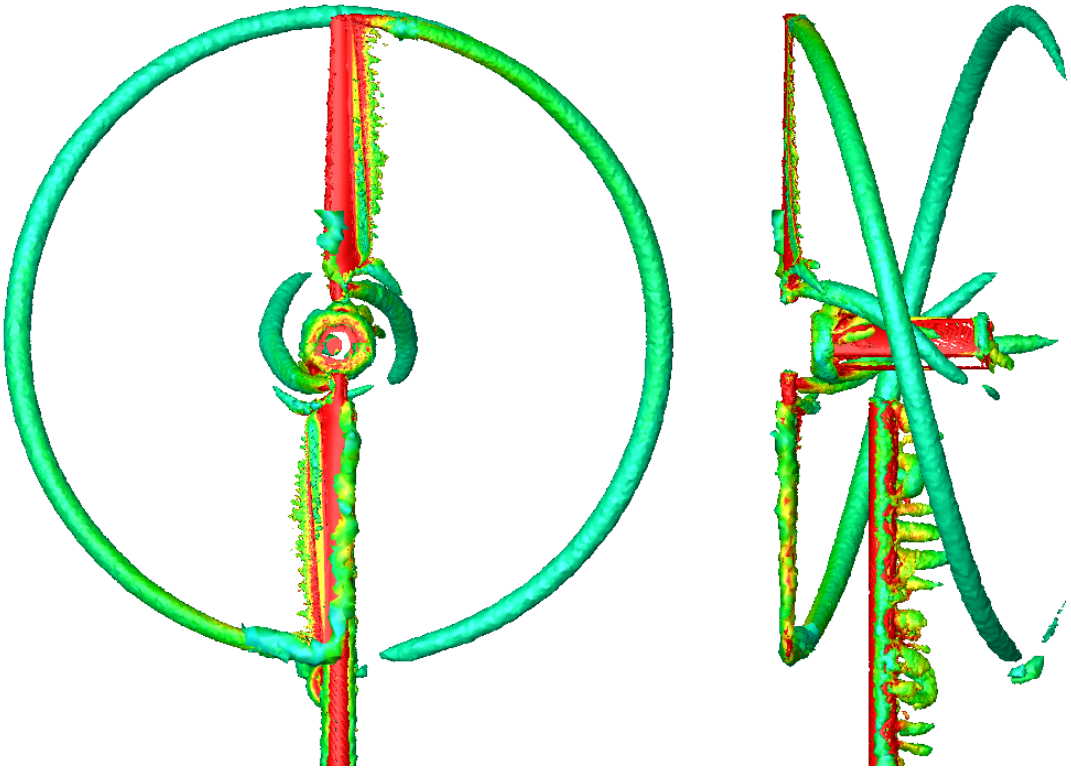
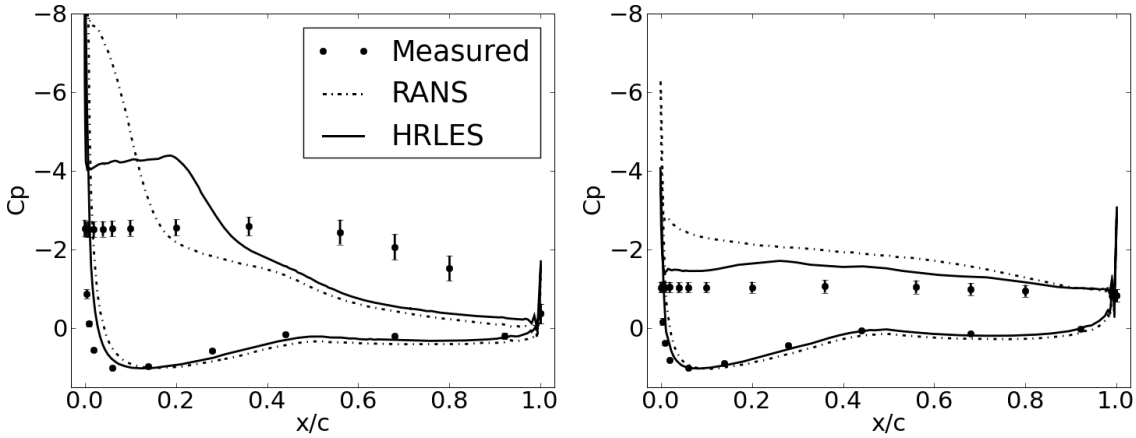


Figure 44: Q-criterion iso-surface ($Q = 0.1$) at 10 m/s, predicted using HR-LES.

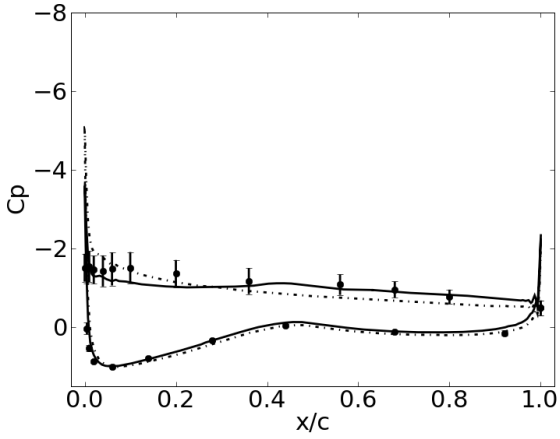
stalled, and the flow is separated everywhere but the tip. At all stations except the tip, HR-LES yields a more accurate prediction of the mean pressure. At the three mid-span stations, HR-LES predicts the experimentally observed stalled flow where RANS does not. Neither turbulence model correctly predicts the experimental pressure at $r/R = 0.3$. HR-LES captures the basic character of the stalled and separated leading edge, but reattaches near 20% chord rather than 55%. This may again be due to a freestream turbulence intensity substantially lower than the experimental value. For this case, RANS was unable to trim due to consistent over-prediction of suction at $r/R = 0.3$, 0.47, and 0.8, and trimming ended at 1° tip pitch. HR-LES, on the other hand, was able to predict the target thrust within 1% using a tip pitch of 4°.

Wake visualizations for the 15 m/s case are shown in Figures 46 and 47. Clearly, HR-LES is able to predict the massive separation from the suction surfaces, as exhibited by the strong vortices visible in Figure 47, shedding from mid-span and persisting for over 90° of wake age. In the RANS solution, some vorticity is shed into the wake from mid-span, but

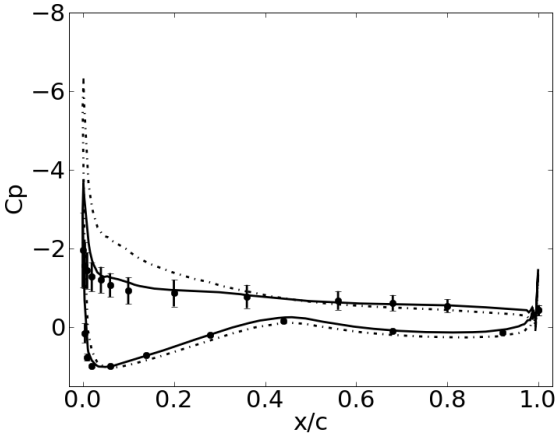


(a) $r/R = 0.30$

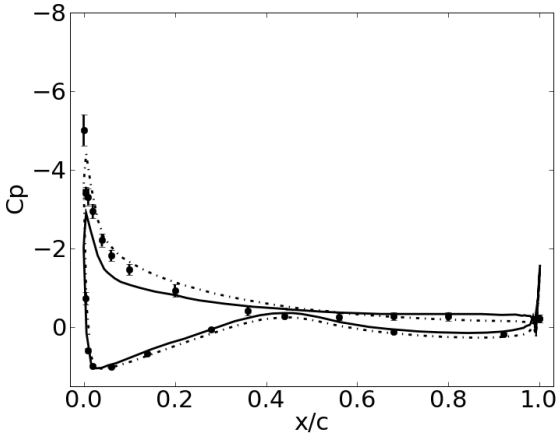
(b) $r/R = 0.47$



(c) $r/R = 0.63$



(d) $r/R = 0.80$



(e) $r/R = 0.95$

Figure 45: Sectional pressure coefficient at 15 m/s, 0 yaw.

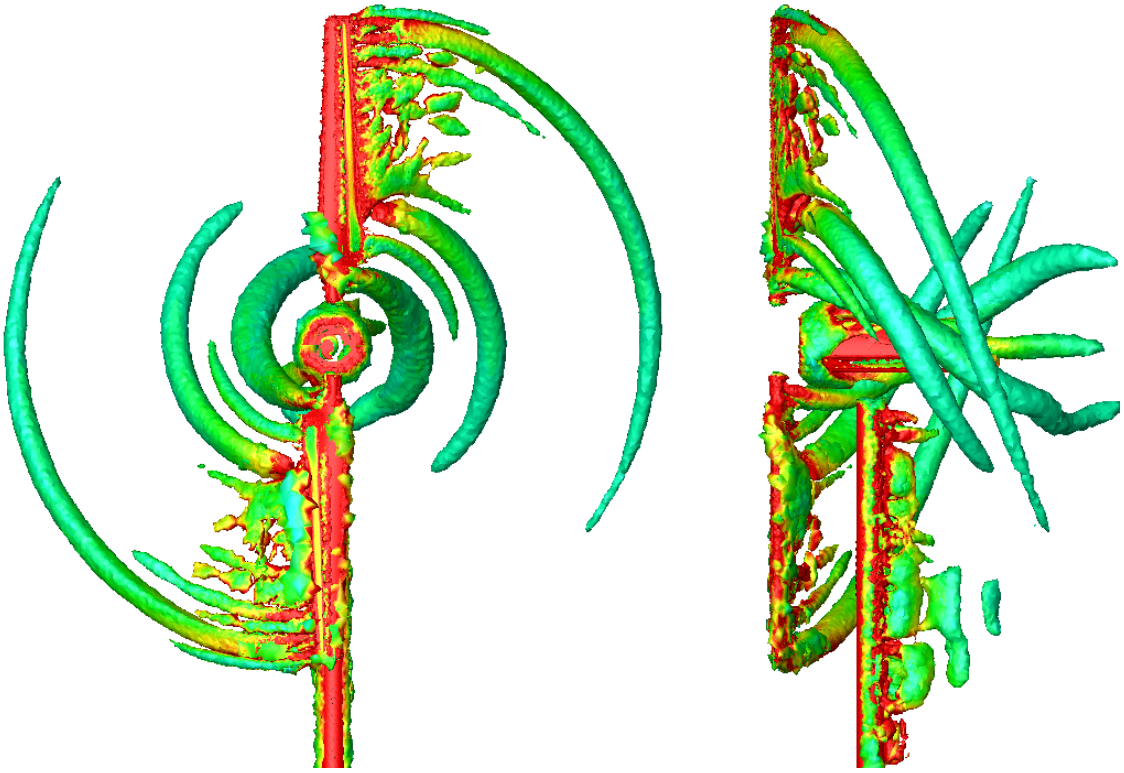


Figure 46: Q-criterion iso-surface ($Q = 0.1$) at 15 m/s, predicted using RANS.

it dissipates much faster than with HR-LES. Both models capture well-defined tip and root vortices, but those vortices are smoother in the RANS solution. In contrast, in the HR-LES solution, the tip vortices interact with the separated wake as they convect downstream, giving the tip vortices a somewhat "twisted" appearance.

8.3.3 15 m/s, 30 degrees yaw

One case was run at 15 m/s and 30° yaw. Since yawed cases are always unsteady, no matter the wind speed, only the HR-LES turbulence model was used here. Mean pressure coefficients are shown in Figure 48. Outboard pressure predictions are observed to be very accurate when correlated with the measured data. At $r/R = 0.3$ and $r/R = 0.47$, the flow is considerably less steady, as evidenced by the size of the bars on the measured data in Figure 48. The predicted pressure is within a standard deviation of the measured mean at most pressure tap locations. With such a chaotic wake, some disagreement is to be expected in mean pressure since the computational results have been averaged over a single revolution rather than the 36 revolutions used to obtain the experimental data.

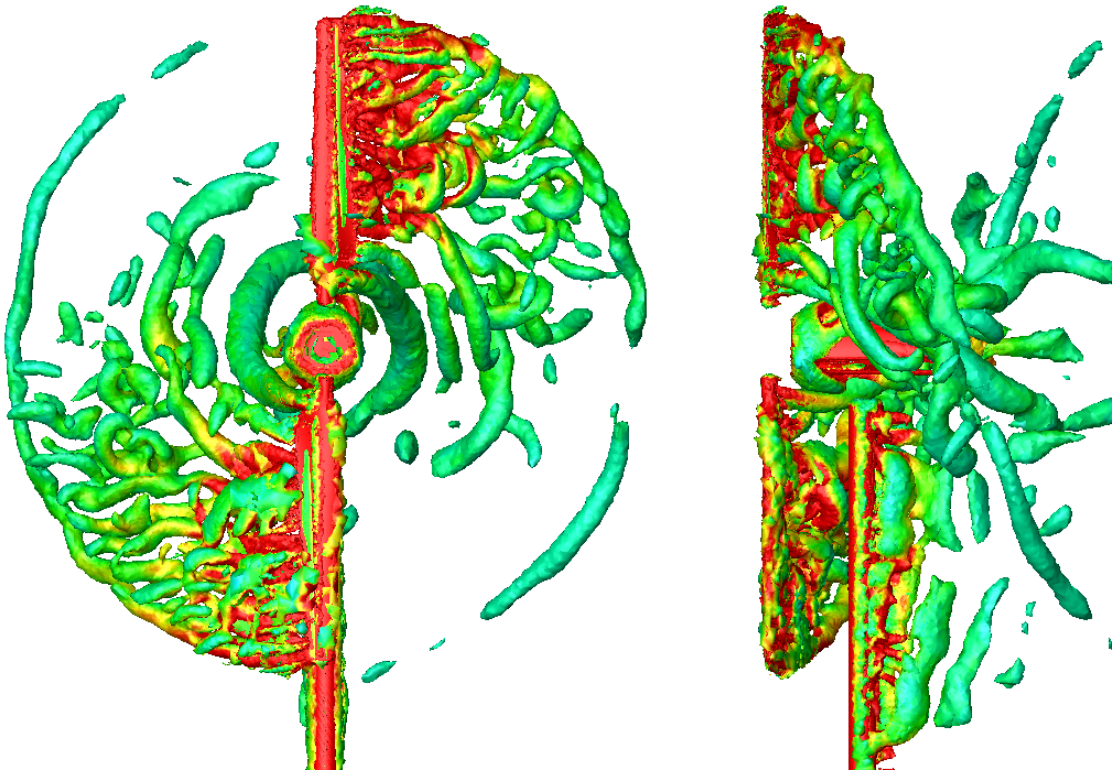
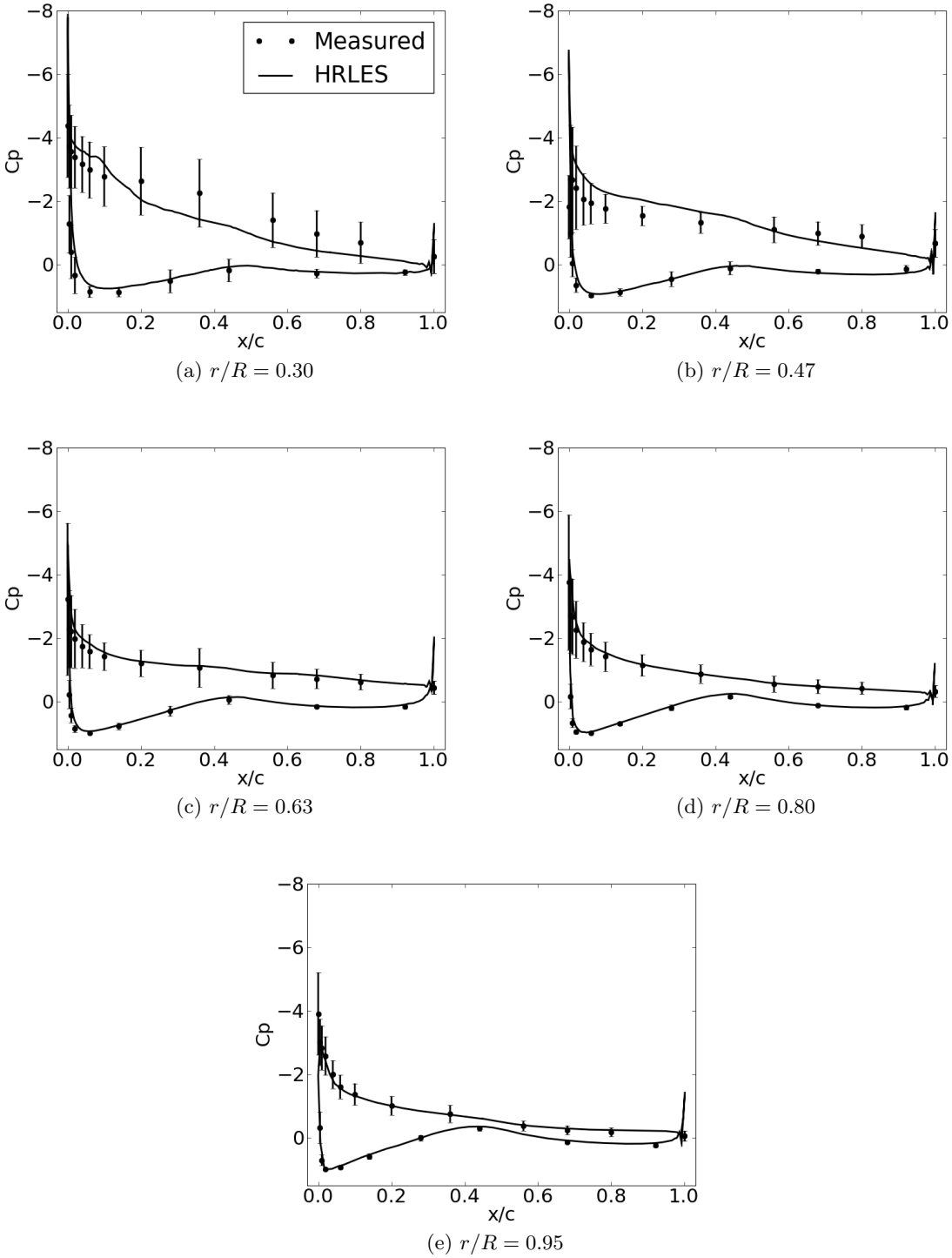


Figure 47: Q-criterion iso-surface at 15 m/s, predicted using HR-LES.

Integrated hub loads vs. azimuth for the 15 m/s, 30° yaw case are shown in Figure 49. The current methodology predicts thrust and torque within 4% and 7%, respectively, of the targets. The unsteadiness observed in the computed thrust between 0° and 180° azimuth is comparable to the wide spread in the measured data, which was gathered over 36 revolutions. In addition to a larger apparent angle of attack, this region is in the shadow of the nacelle, and in the experimental data, the hub instrumentation box and boom. The same can be seen in root flap bending moment, which is also derived from sectional normal force, but with greater weight given to the tip. The maximum torque is somewhat over-predicted, which could be due to the aforementioned transition effect [41]. The maximum torque, and also appears phase shifted by about 60°.

Sectional normal force and pitching moment vs azimuth are shown in Figures 50 and 51. Similar to the integrated loads, the wide spread in the measured data in the downwind half of the rotor is well-predicted. At very high yaw angles, the instrumentation box and camera boom mounted in front of the hub (see Figure 41) have a significant effect on blade pressure



(a) $r/R = 0.30$

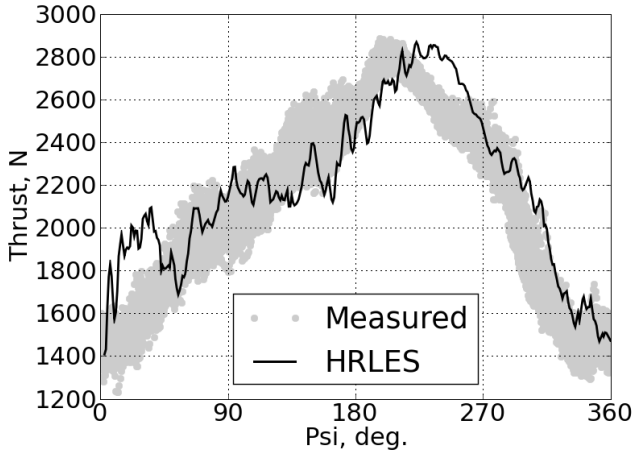
(b) $r/R = 0.47$

(c) $r/R = 0.63$

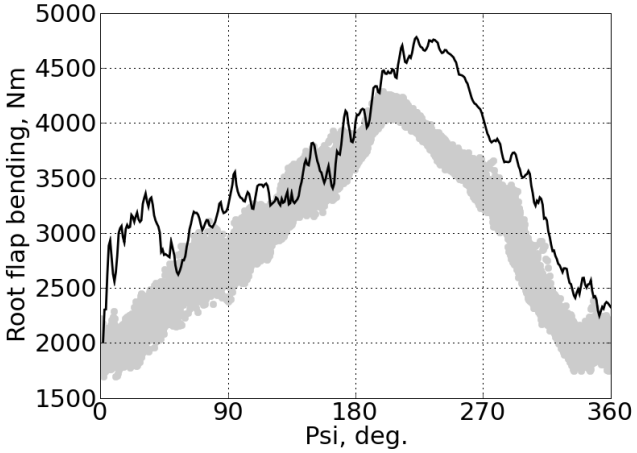
(d) $r/R = 0.80$

(e) $r/R = 0.95$

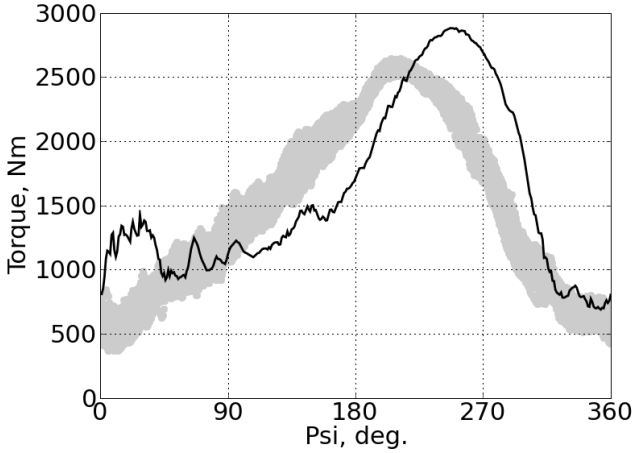
Figure 48: Sectional pressure coefficient at 15 m/s, 30° yaw.



(a) Thrust



(b) Root flap bending



(c) Torque

Figure 49: Integrated hub loads (pressure component only) vs. azimuth at 15 m/s, 30° yaw.

measurements in the $0 \leq \psi < 120°$ region [35] (ψ is zero for a blade at the 12 o'clock position and increases for counter-clockwise rotation when viewed from upstream). However, at 30° yaw, only $r/R < 0.22$ (approximately) is in the box's shadow. The boom's shadow extends to about $r/R < 0.34$. Therefore, only the first pressure tap station, at $r/R = 0.3$ should be affected by the omission of the box and boom in the current simulations. Any unsteadiness at the outer four locations is due to the changing local angle of attack.

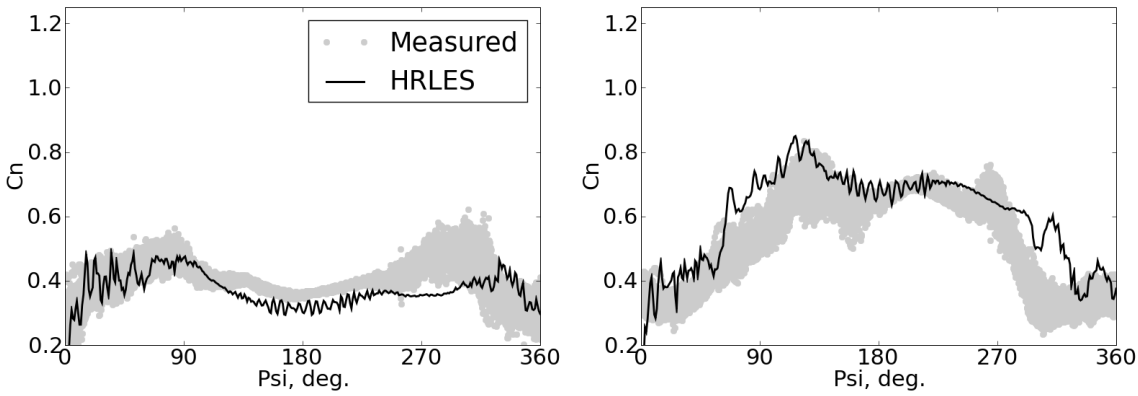
The instantaneous wake visualizations in Figure 52 clearly illustrate the asymmetry expected in a yawed case. A larger extent of periodic shedding can be observed for the blade that is in the 6 o'clock position than for the blade at 12 o'clock. The $90° \leq \psi < 270°$ region experiences a higher incident velocity than the other half of the rotor disc. Therefore, the local angle of attack is higher, leading to greater separation.

8.3.4 Compressible solutions

Using an incompressible solution algorithm was essential to the results presented here. Figure 53 compares the wake presented earlier for the 15 m/s, 0° yaw case, to the same case computed with a compressible method [105] (without low-Mach preconditioning, but using HR-LES). The compressible algorithm is much more dissipative at this Mach number, leading to less well-defined tip and root vortices. The same is true of the mid-span separation vortices. An incompressible method also improves computational throughput; the energy equation need not be solved, so the mean flow Jacobian consists of 4×4 rather than 5×5 blocks.

8.4 Performance

All flow solutions presented here were run on the Intel 64 Cluster "Abe" at the National Center for Supercomputing Applications (NCSA). Each cluster node had one 2.33 GHz quad-core CPU. As in the HART simulations, 400 cores were used for the flow solution and a single core for hole-cutting. At 720 steps per revolution, one revolution took 25–30 hours of wall time. Trimming required 3–4 revolutions. Fifteen percent of the total solution time was spent in hole-cutting and overset assembly.

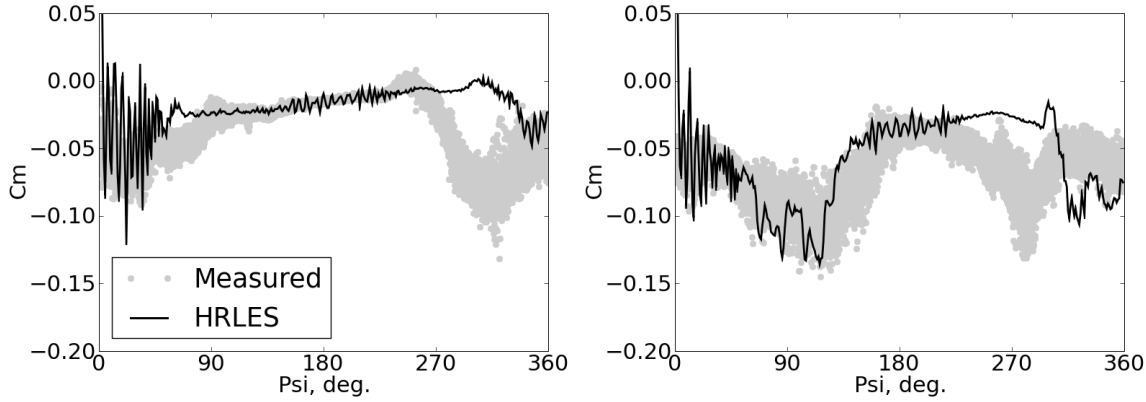


(a) $r/R = 0.30$ (b) $r/R = 0.47$

(c) $r/R = 0.63$ (d) $r/R = 0.80$

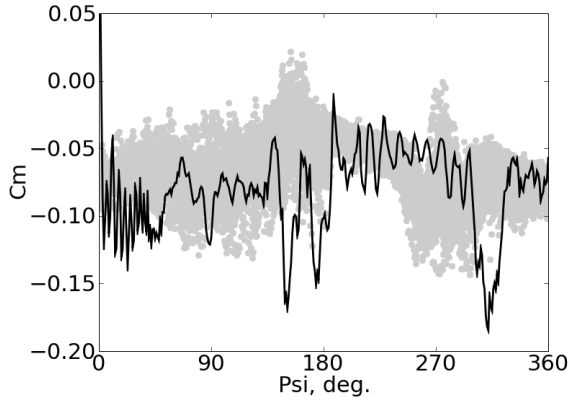
(e) $r/R = 0.95$

Figure 50: Sectional normal force coefficient, normalized by tip dynamic pressure, vs. azimuth at 15 m/s, 30° yaw.

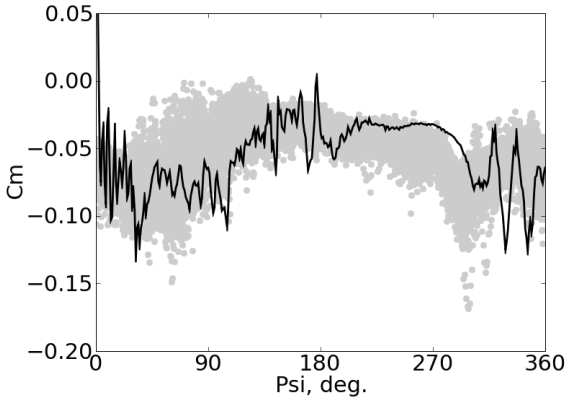


(a) $r/R = 0.30$

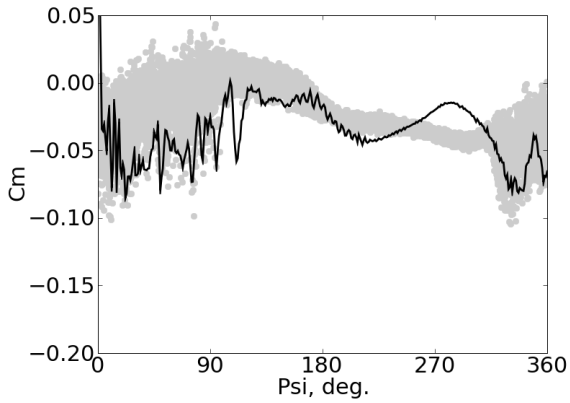
(b) $r/R = 0.47$



(c) $r/R = 0.63$



(d) $r/R = 0.80$



(e) $r/R = 0.95$

Figure 51: Sectional pitching moment coefficient, normalized by tip dynamic pressure, vs. azimuth at 15 m/s, 30° yaw.

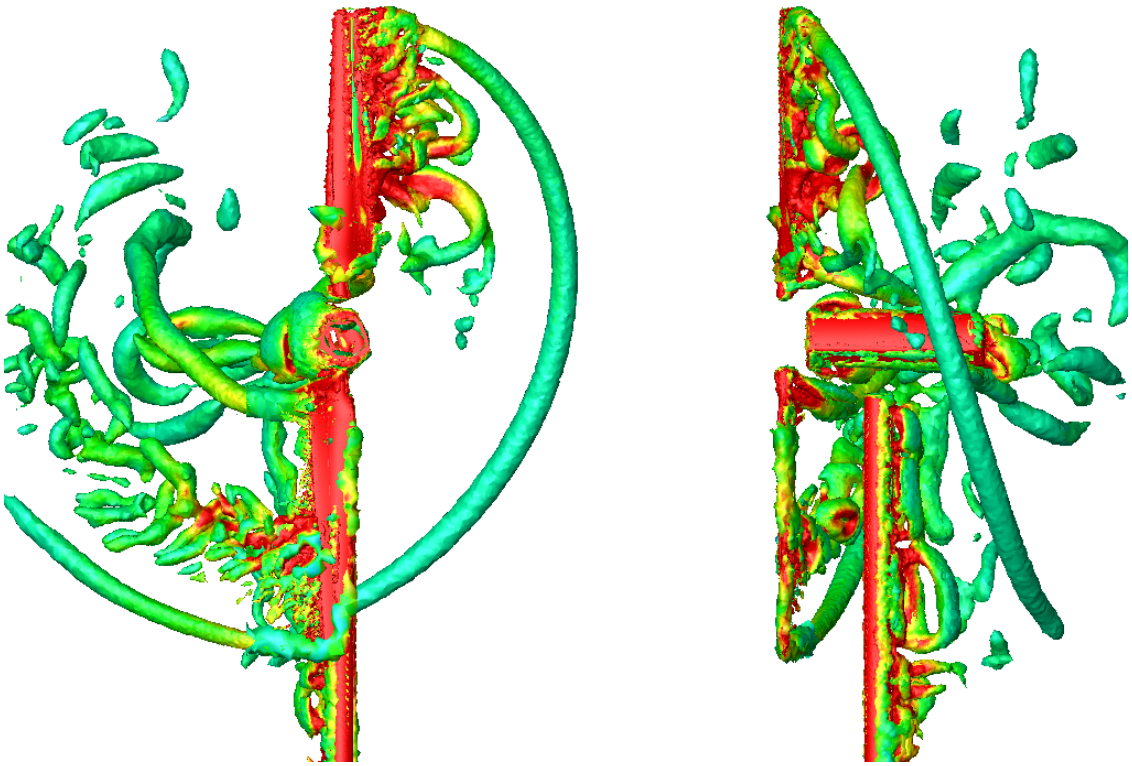
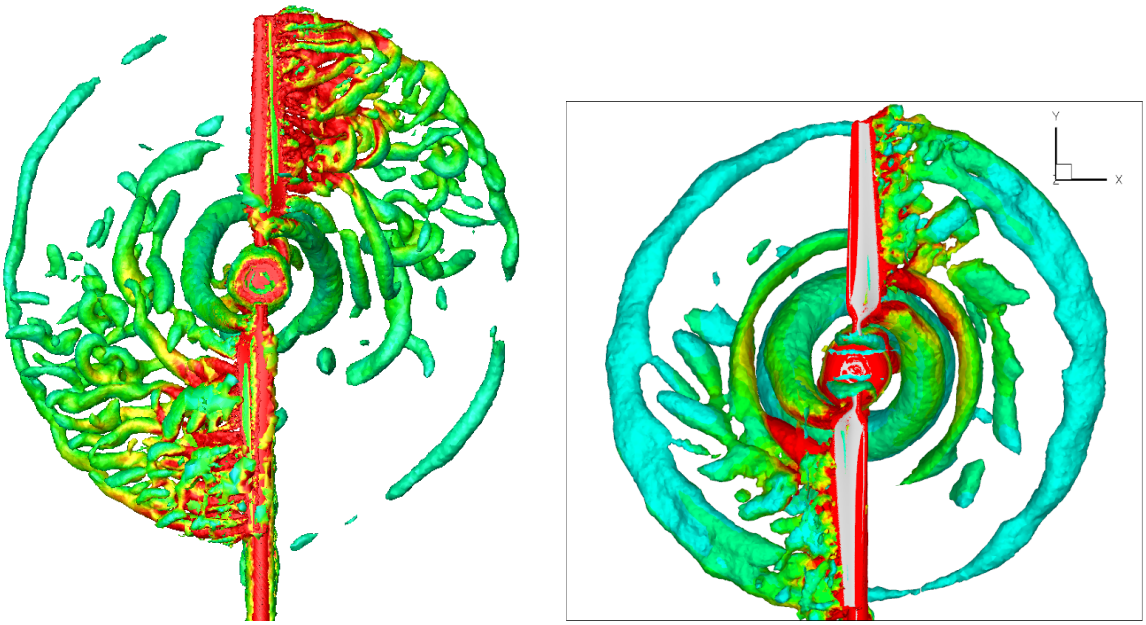


Figure 52: Q-criterion iso-surface ($Q = 0.1$) at 15 m/s and 30° yaw, predicted using HR-LES.



(a) Incompressible

(b) Compressible, $M_{ref} = M_{tip} = 0.11$

Figure 53: Q-criterion iso-surface at 15 m/s and 0° yaw, computed using a compressible algorithm [105].

CHAPTER IX

CONCLUSIONS

A range of different CFD techniques of varying accuracy and computational expense has been developed for simulating horizontal axis wind turbines. Improved unsteady actuator blade methods for efficiently simulating wind turbine wakes without fully resolving the blades were developed. Possible applications for these methods include turbine-to-turbine interaction studies. A hybrid RANS/LES turbulence model was modified for unstructured methodologies and evaluated on several configurations of interest, including a cylinder in cross-flow and flatback airfoil. A tight CFD/CSD coupling methodology for very high fidelity overset simulations of wind turbines was demonstrated. Unstructured overset simulations of a wind turbine using a hybrid RANS/LES turbulence model were performed and correlated with an experimental dataset. All overset turbine simulations were conducted in a time-accurate fashion in an inertial reference frame and included the turbine tower and nacelle. The following conclusions have been drawn.

For the improved actuator blade methods:

- An improved nearest neighbor search algorithm based on kd-trees was shown to decrease the CPU time for source-to-node association by as much as 85% when there is a large number of mesh nodes per CPU. This can result in a 20% reduction in total wall time, with a minimal impact on memory usage.

- It was also shown that using the actual non-linear twist distribution rather than a linear approximation of the twist was necessary for accurate actuator wake predictions.

- Actuator blade simulations of the NREL Phase VI Unsteady Aerodynamics Experiment at 7 m/s were compared against high-fidelity overset simulations that model the real blade geometry. It was shown that the actuator blade method predicts tip and root vortex trajectories nearly identical to the overset method in the near wake with

a 38% reduction in CPU time per degree of freedom compared to the overset solution.

For hybrid RANS/LES turbulence modeling:

- The HRLES model yields improved predictions of drag and shedding frequency for bluff-body flow when compared to its underlying two-equation RANS model.
- These models correctly predict the unsteady, 3-D, and chaotic nature of bluff-body wakes.
- The HRLES model developed here is sensitive to mesh cells with a very high aspect ratio. Non-physical solutions were observed for meshes containing tetrahedra with aspect ratios greater than 8 close to the wall but outside the boundary layer. Most unstructured grid generators automatically generate low aspect ratio cells outside the boundary layer (they are unavoidable *in* the boundary layer), but trouble can arise in the special case of extruded meshes.

Pertaining to the high fidelity aeroelastic simulations:

- Just as tight coupling can be used to simulate maneuvering flight in helicopters [66, 2], the process can be applied to with turbine simulations that include yaw adjustments and momentary wind gusts, neither of which have a quasi-periodic loading as is required for a loose coupling simulation.
- Though the simulation used to demonstrate tight CFD/CSD coupling was carried out using a helicopter rotor for which an experimental dataset was available for verification, the principles are similar for a wind turbine rotor (see future work below).
- Should an experimental dataset for a wind turbine rotor aeroelastic effects more typical of a utility-scale turbine become publicly available in the future, this coupling methodology can then be demonstrated for that rotor without any modifications to the algorithm.

For overset wind turbine simulations:

- At 10 m/s wind speed, RANS and HRLES methods were found to yield nearly identical solutions as the boundary layer remains attached over most of the rotor.

- Poor prediction of the peak suction at $r/R = 0.47$ in the 10 m/s case caused both RANS and HRLES to over-predict rotor thrust and torque, precluding a trimmed solution. Sensitivity of this rotor to turbulent inflow conditions was suggested as an explanation for this discrepancy.

- At 15 m/s, where the rotor is almost fully stalled, HRLES was able to predict thrust within 1% of the experimentally measured value. Where RANS over-predicted torque by 51%, HRLES predicted torque within 16% of the measured value. The improved loads predictions were due to the ability of HRLES to capture the highly unsteady 3-D wake of the stalled blades.

- HRLES was also able to accurately capture blade loads in yawed flow with excellent mean C_p correlations across the span. Wake asymmetry was accurately captured as well as increased unsteadiness in sectional loads in the downwind half of the rotor disc.

9.1 Recommendations for Future Work

Future work in improved actuator blade methods could include:

- Finite state unsteady aerodynamics

  A shortcoming of the actuator blade methodology is that the source terms that drive the CFD solution are based on extremely simple linear, quasi-steady aerodynamics. Much more accurate approximations for unsteady aerodynamics exist in the form of finite state aerodynamics algorithms, such as those of Peters [72]. These methods are still very efficient compared to full overset CFD. Since they already exist in many comprehensive codes like DYMORE, coupling with a comprehensive code could introduce these unsteady effects with relatively little effort in implementation. The CFD code would provide local angles of attack to the comprehensive code, which would then provide instantaneous lift and drag, which can then be used to compute the source

terms. Coupling of a steady-state CFD actuator disc with a comprehensive code was previously demonstrated by Renaud et al. [78], so this would be a natural extension to time-accurate simulations.

- Non-rectangular planforms

  Presently, the actuator blade method does not allow for non-rectangular planforms, but modern wind turbine blades are typically tapered. Furthermore, novel designs, such as those with sweep twist adaptive rotors (STAR) [7] may have blades that are curved and swept as well as tapered. Due to the rectangular assumption, sources will not be located in the correct locations to properly approximate the real rotor. However, it would be simple to accomodate general chord distributions in the same way that general twist distributions are now supported.

- Advanced actuator blade applications

  Further applications of actuator blades should be explored, in particular, interaction studies between groups of turbines or ground terrain.

Future work in developing the HRLES turbulence model could include:

- Other test cases at super-critical Reynolds numbers should be conducted to verify that the low Reynolds number was responsible for many of the disagreements with experimental data in the cylinder wake.

- The HRLES model should be extended to utilize the dynamic closure coefficients specified in the LDKM model. This would make the model more generally applicable and not as susceptable to grid-related dissipation.

- The characteristic grid length used in the LES destruction term should be revisited. Spalart [100] notes that on unstructured grids, (dual) cell diameter would be a more appropriate estimate of mesh size than volume$^{1/3}$. Isotropic LES models, like the one used in the hybrid model here, have no preferred eddy orientation, so characteristic length should ideally be the largest distance between a node and its neighbors in any direction.

- The flatback airfoil simulations with tunnel walls should be revisited once the wall porosity is known. Given the large tunnel blockage, modeling the solid upper and lower walls can be expected to improve correlations with measured data but only if a porous boundary condition or empirical correction is used in modeling the acoustic screens in the side walls.

Possible areas for future development of overset wind turbine simulation techniques include:

- Turbulent transition prediction

  Since many wind turbines are designed with laminar flow airfoils, and do indeed have regions of laminar flow in normal operation, a means of predicting boundary layer transition would improve loads predictions in critical cases. Transition prediction could range from simple methods like Michel's criterion [57] to high-fidelity methods like Menter's γ-$\widetilde{\mathrm{Re}}_\theta$ model [56, 97], which requires computing another system of transport equations. Of course, the computational cost of the latter approach would be considerable, on the same order as adding a two-equation turbulence model to a laminar solver.

- Rigid trim procedures

  Algorithms for computing the trimmed state of a rigid rotor would be substantially reduce user burden in overset rotor cases.

- More efficient overset strategies

  The cost of overset assembly is considerable, on the order of 25%. Strategies to reduce this cost would dramatically improve performance of overset simulations. Improvements could include parallel hole-cutting algorithms, though hole-cutting has proved resistant to attempts at acceleration in the past [64]. Optimization of the mesh partitioning strategy for rotor simulations could trade some solver inefficiency for improved parallel performance in hole-cutting. Rotor-specific partition strategies have already been explored by Noack [64], but their effect on solver efficiency has not

been determined.

- Wind turbine applications of tight coupling

  As more experimental and field datasets become available, they should be used to provide further correlations for evaluating the tight coupling methods developed here. Predictions of structural responses to gusts and the vertical shear profile would be particularly relevant to the wind energy community. Complicating factors unique to wind turbines, such as flexible towers, non-uniform freestream, and non-uniform blade construction should also be investigated so that CFD/CSD coupling algorithms can account for their effects.

- Atmospheric boundary layer effects

  Both the approximate and high-fidelity prediction methods developed here could be used to explore the effects of wind shear in the atmospheric boundary layer on rotor performance and fatigue.

- Fundamental wake characterization

  The basic physical processes in the development of wind turbine wakes should be studied with an emphasis on how turbine wakes differ from helicopter wakes. Though there are obvious similarities between helicopter and wind turbine rotors, there should be fundamental differences in how their wakes develop. This wake characterization will require the application of higher-order algorithms and/or finer grid resolution (via either uniform or adaptive mesh refinement) to resolve tip vortices at long wake age.

REFERENCES

[1] "20% Wind Energy by 2030," Tech. Rep. DOE/GO-102008-2567, U.S. Department of Energy, July 2008.

[2] Abhishek, A., Datta, A., and Chopra, I., "Comprehensive Analysis, Prediction, and Validation of UH-60A Blade Loads in Unsteady Maneuvering Flight," in *American Helicopter Society 63rd Annual Forum*, (Virginia Beach, VA), May 1-3, 2007.

[3] Abras, J. N., *Enhancement of Aeroelastic Rotor Airload Prediction Methods*. PhD thesis, Georgia Institute of Technology, 2009.

[4] Abras, J. N., Lynch, C. E., and Smith, M. J., "Advances in Rotorcraft Simulations with Unstructured CFD," in *American Helicopter Society 63rd Annual Forum*, (Virginia Beach, VA), May 1-3, 2007.

[5] Ahmad, J. and Duque, E., "Helicopter Rotor Blade Computation in Unsteady Flows Using Moving Overset Grids," *Journal of Aircraft*, vol. 33, no. 1, pp. 54–60, 1996.

[6] Anderson, W., Rausch, R., and Bonhaus, D., "Implicit/Multigrid Algorithms for Incompressible Turbulent Flows on Unstructured Grids," *Journal of Computational Physics*, vol. 128, no. 2, pp. 391–408, 1996.

[7] Ashwill, T., Kanaby, G., Jackson, K., and Zuteck, M., "Development of the Sweep-Twist Adaptive Rotor (STAR) Blade," in *48th AIAA Aerospace Sciences Meeting and Exhibit*, (Orlando, FL), AIAA-2010-1582, Jan. 4-7, 2010.

[8] Bauchau, O. A. and Ahmad, J. U., "Advanced CFD and CSD Methods for Multidisciplinary Applications in Rotorcraft Problems," in *6th AIAA, NASA, and ISSMO Symposium on Multidisciplinary Analysis and Optimization*, (Bellevue, WA), AIAA-1996-4151, Sept. 4-6, 1996.

[9] Bauchau, O. July 2010. Private communication.

[10] Bauchau, O. A., Bottasso, C. L., and Nikishkov, Y. G., "Modeling Rotorcraft Dynamics with Finite Element Multibody Procedures," *Mathematical and Computer Modeling*, vol. 33, no. 10-11, pp. 1113–1137, 2001.

[11] Bauchau, O.A., "Dymore User's Manual," tech. rep., Georgia Institute of Technology, June 2010.

[12] Baurle, R. A., Tam, C. J., Edwards, J. R., and Hassan, H. A., "Hybrid Simulation Approach for Cavity Flows: Blending Algorithm, and Boundary Treatment Issues," *AIAA Journal*, vol. 41, no. 8, pp. 1463–1480, 2003.

[13] Bentley, J. L., "Multidimensional Binary Search Trees Used for Associate Searching," *Communications of the ACM*, vol. 18, no. 9, pp. 509–517, 1975.

[14] Berg, D. and Zayas, J., "Aerodynamic and Aeroacoustic Properties of Flatback Airfoils," in *46th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2008-1455, Jan. 5-8, 2008.

[15] Biedron, R., Vatsa, V., and Atkins, H., "Simulation of Unsteady Flows Using an Unstructured Navier-Stokes Solver on Moving and Stationary Grids," in *23rd AIAA Applied Aerodynamics Conference*, (Toronto, Canada), AIAA-2005-5093, June 6-9, 2005.

[16] Biedron, R. T. and Lee-Rausch, E. M., "Rotor Airloads Prediction Using Unstructured Meshes and Loose CFD/CSD Coupling," in *26th AIAA Applied Aerodynamics Conference*, (Honolulu, HI), AIAA-2008-7341, Aug. 18-21, 2008.

[17] Bonhaus, D., "An Upwind Multigrid Method For Solving Viscous Flows On Unstructured Triangular Meshes," Master's thesis, George Washington University, 1993.

[18] Bridgeman, J. O., Strawn, R. C., Caradonna, F. X., and Chen, C. S., "Advanced Rotor Computations with a Corrected Potential Method," in *American Helicopter Society 45th Annual Forum Proceedings*, (Boston, MA), May 22-24 1989.

[19] Buning, P., Parks, S., Chan, W., and Renze, K., "Application of the Chimera Overlapped Grid Scheme to Simulation of Space Shuttle Ascent Flows," in *Proceedings of the Fourth International Symposium on Computational Fluid Dynamics*, vol. 1, (Davis, CA), pp. 132–137, Sept. 1991.

[20] Cantaloube, B. and Huberson, S., "A New Approach Using Vortex Point Method for Prediction of Rotor Performance in Hover and Forward Flight," *Vertica*, vol. 10, no. 2, pp. 187–200, 1986.

[21] Catlett, C. et al., "*TeraGrid*: Analysis of organization, system architecture, and middleware enabling new types of applications," in *HPC and Grids in Action, Ed. Lucio Grandinetti, IOS Press 'Advances in Parallel Computing' series*, (Amsterdam), 2007.

[22] Chao, D. D. and van Dam, C. P., "RaNS Analysis of an Inboard Flatback Modification to the NREL Phase VI Rotor," in *44th Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2006-0195, Jan. 9-12, 2006.

[23] Chorin, A., "A Numerical Method for Solving Incompressible Viscous Flow Problems," *Journal of Computational Physics*, vol. 2, no. 1, pp. 12–26, 1967.

[24] Cockle, R., "Wind Whips Up Health Fears," *The Oregonian*, Aug. 10, 2008.

[25] Coton, F. N., Wang, T., and Galbraith, R. A. M., "An Examination of Key Aerodynamic Modeling Issues Raised by the NREL Blind Comparison," in *21st ASME Wind Energy Symposium and 40th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2002-0038, Jan. 14-17, 2002.

[26] Datta, A., Sitaraman, J., Baeder, J. D., and Chopra, I., "Analysis Refinements for Prediction of Rotor Vibratory Loads in High Speed Forward Flight," in *American Helicopter Society 60th Annual Forum Proceedings*, (Baltimore, MD), June 7-10, 2004.

[27] Davidson, P. A., *Turbulence: An Introduction for Scientists and Engineers*. New York: Oxford University Press, 2004.

[28] Duque, E. P. N., Burklund, M. D., and Johnson, W., "Navier-Stokes and Comprehensive Analysis Performance Predictions of the NREL Phase VI Experiment," *Journal of Solar Energy Engineering*, vol. 125, pp. 457–467, November 2003.

[29] Duque, E. P. N., van Dam, C. P., and Hughes, S. C., "Navier-Stokes Simulations of the Combined Experiment Phase II Rotor," in *ASME Wind Energy Symposium*, (Reno, NV), AIAA-1999-0037, Jan. 11-14, 1999.

[30] Duque, E. et al., "Revolutionary Physics-Based Design Tools for Quiet Helicopters," in *44th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2006-1068, Jan. 9-12, 2006.

[31] Elliott, D. L., Holladay, C. G, Barchet, W. R., Foote, H. P.,Sandusky, W. F., "Wind Energy Resource Atlas of the United States," Tech. Rep. DOE/CH 10093-4, Pacific Northwest Laboratory, Richland, Washington, Oct. 1986.

[32] Friedman, J. H., Bentley, J. L., Finkeyl, R. A., "An Algorithm for Finding Best Matches in Logarithmic Expected Time," Tech. Rep. STAN-CS-75-482, Stanford CS Rep., Feb. 1975.

[33] Gaither, J. A., Marcum, D. L., and Mitchell, B., "Solidmesh: A Solid Modeling Approach to Unstructured Grid Generation," in *7th International Conference on Numerical Grid Generation in Computational Field Simulations*, (Whistler, Canada), Sept. 25-28, 2000.

[34] Grosveld, F. W., "Prediction of Broadband Noise from Horizontal Axis Wind Turbines," *Journal of Propulsion and Power*, vol. 1, no. 4, pp. 292–299, 1985.

[35] Hand, M. M. et al., "Unsteady Aerodynamics Experiment Phase VI: Wind Tunnel Test Configurations and Available Data Campaigns," Tech. Rep. NREL/TP-500-29955, National Renewable Energy Laboratory, Golden, CO, Dec. 2001.

[36] Huyer, S. A., Simms, D., Robinson, M. C., "Unsteady Aerodynamics Associated with a Horizontal-Axis Wind Turbine," *AIAA Journal*, vol. 34, no. 7, pp. 1410–1419, 1996.

[37] Innovative Strategies for Rotary-Wing Coupled Aeroelastic Simulations, "NASA Langley Research Center." Award No. NNX07AP43A, August 15, 2007 to February 14, 2011.

[38] Ivanell, S., Sørensen, J. N., Mikkelsen, R., and Henningson, D., "Analysis of Numerically Generated Wake Structures," *Wind Energy*, vol. 12, no. 1, pp. 63–80, 2009.

[39] Kim, W.-W. and Menon, S., "An Unsteady Incompressible Navier-Stokes Solver for Large Eddy Simulation of Turbulent Flows," *International Journal for Numerical Methods in Fluids*, vol. 31, no. 6, pp. 983–1017, 1999.

[40] Kravchenko, A. G. and Moin, P., "Numerical Studies of Flow Over a Circular Cylinder at $Re_D = 3900$," *Physics of Fluids*, vol. 12, no. 2, pp. 403–417, 2000.

[41] Langtry, R., Gola, J., and Menter, F., "Predicting 2D Airfoil and 3D Wind Turbine Rotor Performance Using a Transition Model for General CFD Codes," in *44th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2005-0395, Jan. 9-12, 2006.

[42] Le Chuiton, F., "Actuator Disc Modeling for Helicopter Rotors," *Aerospace Science and Technology*, vol. 8, no. 4, pp. 285–297, 2004.

[43] Le Pape, A. and Lecanu, J., "3D Navier-Stokes Computations of a Stall-regulated Wind Turbine," *Wind Energy*, vol. 7, no. 4, pp. 309–324, 2004.

[44] Lee, C. S., Saberi, H., and Ormiston, R. A., "Aerodynamic and Numerical Issues for Coupling CFD into Comprehensive Rotor Analysis," in *American Helicopter Society 53rd Annual Forum Proceedings*, (Virginia Beach, VA), April 29–May 1, 1997.

[45] Leishman, J. G., "Challenges in Modeling the Unsteady Aerodynamics of Wind Turbines," in *21st ASME Wind Energy Symposium and 40th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2002-0037, Jan. 14-17, 2002.

[46] Lim, J. W., Nygaard, T. A., Strawn, R., and Potsdam, M., "BVI Airloads Prediction Using CFD/CSD Loose Coupling," in *AHS San Francisco Bay Area Chapter Specialists' Conference on Vertical Lift Aircraft Design*, (San Francisco, CA), Jan. 18-20, 2006.

[47] Lim, J. and Strawn, R., "Prediction of HART II Rotor BVI Loading and Wake System Using CFD/CSD Loose Coupling," in *45th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2007-1281, Jan. 8-11, 2007.

[48] Lourenco, L.M. and Shih, C., "Characteristics of the Plane Turbulent Near Wake of a Circular Cylinder." Digitized from Figures 13, 15, and 22 in Reference [40].

[49] Madsen, H. and Johansen. J. and Sørensen, N. and Larsen, G. and Hansen, M., "Simulation of Low Frequency Noise from a Downwind Wind Turbine Rotor," in *45th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2007-0623, Jan. 8-11, 2007.

[50] Makinen, S., Hill, M., Gandhi, F., Long, L. N., Vasilescu, R., and Sankar, L., "A Study of the HART-I Rotor with Loose Computational Fluid/Structural Dynamic Coupling," in *American Helicopter Society 62nd Annual Forum Proceedings*, (Phoenix, AZ), May 9-11, 2006.

[51] Marcum, D. L., "Unstructured Grid Generation Using Automatic Point Insertion and Local Reconnection," in *The Handbook of Grid Generation* (Thompson, J. F., Soni, B., and Weatherill, N. P., eds.), ch. 18, CRC Press, 1998.

[52] Marcum, D. L. and Weatherill, N. P., "Unstructured Grid Generation Using Iterative Point Insertion and Local Reconnection," *AIAA Journal*, vol. 33, no. 9, pp. 1619–1625, 1995.

[53] Meakin, R., "Moving Body Overset Grid Methods for Complete Aircraft Tiltrotor Simulations," in *11th AIAA Computational Fluid Dynamics Conference*, (Orlando, FL), AIAA-1993-3350, July 6-9, 1993.

[54] Menter, F. R., "Improved Two-Equation k-ω Turbulence Models for Aerodynamic Flows," Tech. Rep. NASA TM 103975, NASA, Oct. 1992.

[55] Menter, F., "Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications," *AIAA Journal*, vol. 32, no. 8, pp. 598–605, 1994.

[56] Menter, F. et al., "Correlation-Based Transition Model Using Local Variables—Part I: Model Formulation," *Journal of Turbomachinery*, vol. 128, no. 3, pp. 413–422, 2006.

[57] Michel, R., "Etude de la Transition sur les Profiles d'Aile; Establissement d'un Critère de Determination de Point de Transition et Calcul de la Trainée de Profile Incompressible," Tech. Rep. 1/1578A, ONERA, 1951.

[58] Mikkelsen, R., *Actuator Disc Methods Applied to Wind Turbines*. PhD thesis, Technical University of Denmark, 2003.

[59] Montaudouin, J., Heo, S., Smith, M., and Bauchau, O., "Aerodynamic and Aeroelastic Analysis of Rotors at High Advance Ratio," in *36th European Rotorcraft Forum*, (Paris, France), Sept. 7-9, 2010.

[60] Moriarty, P.J., "AeroDyn Theory Manual," Tech. Rep. NREL/EL-500-36881, National Renewable Energy Laboratory, Dec. 2005.

[61] Narducci, R., "Comparison of Blade Tip Vortex Calculations to Wind Tunnel Measurements," in *Proceedings of the Army Research Office Rotorcraft Wake Prediction Basic Research Workshop*, (Georgia Institute of Technology, Atlanta, GA), Mar. 16-17, 2009.

[62] Noack, R., "DiRTlib: A Library to Add an Overset Capability to Your Flow Solver," in *17th AIAA Computational Fluid Dynamics Conference*, (Toronto, Ontario, Canada), AIAA-2005-5116, June 6-9, 2005.

[63] Noack, R., "SUGGAR: A General Capability for Moving Body Overset Grid Assembly," in *17th AIAA Computational Fluid Dynamics Conference*, (Toronto, Ontario, Canada), AIAA-2005-5117, June 6-9, 2005.

[64] Noack, R., Boger, D., Kunz, R., and Carrica, P., "Suggar++: An Improved General Overset Grid Assembly Capability," in *19th AIAA Computational Fluid Dynamics Conference*, (Austin, TX), AIAA-2009-3992, June 22-25, 2009.

[65] Norberg, C. Unpublished data digitized from Figure 11 in Reference [40].

[66] Nygaard, T., Saberi, H., Ormiston, R., Strawn, R., and Potsdam, M., "CFD and CSD Coupling Algorithms and Fluid Structure Interface for Rotorcraft Aeromechanics in Steady and Transient Flight Conditions," in *American Helicopter Society 62nd Annual Forum Proceedings*, (Phoenix, AZ), May 9-11, 2006.

[67] O'Brien, D. M., *Analysis of Computational Modeling Techniques for Complete Rotorcraft Configurations*. PhD thesis, Georgia Institute of Technology, 2006.

[68] O'Brien, D. M. and Smith, M. J., "Analysis of Rotor-Fuselage Interactions Using Various Rotor Models," in *AIAA 43rd Aerospace Sciences Meeting*, (Reno, NV), AIAA-2005-0468, Jan. 10-13, 2005.

[69] Obrien, D. M. and Smith, M. J., "Understanding the Physical Implications of Approximate Rotor Methods Using an Unstructured CFD Method," in *31st European Rotorcraft Forum*, (Florence, Italy), Sept. 13-15, 2005.

[70] Ong, L. and Wallace, J., "The Velocity Field of the Turbulent Very Near Wake of a Circular Cylinder," *Experimental Fluids*, vol. 20, no. 6, pp. 441–453, 1996.

[71] Paquette, J. A. and Veers, P. A., "Increased Strength in Wind Turbine Blades Through Innovative Structural Design," in *Proceedings of the European Wind Energy Conference*, (Milan, Italy), May 7-10, 2007.

[72] Peters, D. A., Karunamoorthy, S., and Cao, Wen-Ming, "Finite State Induced Flow Models Part I: Two-Dimensional Thin Airfoil," *Journal of Aircraft*, vol. 32, no. 2, pp. 313–321, 1995.

[73] Pirzadeh, S., "Three-Dimensional Unstructured Viscous Grids by the Advancing Front Method," *AIAA Journal*, vol. 34, no. 1, pp. 43–49, 1996.

[74] Potsdam, M. and Mavriplis, D., "Unstructured Mesh CFD Aerodynamic Analysis of the NREL Phase VI Rotor," in *47th AIAA Aerospace Sciences Meeting*, (Orlando, FL), AIAA-2009-1221, Jan. 5-8, 2009.

[75] Potsdam, M., Smith, M., and Renaud, T., "Unsteady Computations of Rotor-Fuselage Interaction," in *35th European Rotorcraft Forum*, (Hamburg, Germany), Sept. 22-25, 2009.

[76] Potsdam, M. A., Venkateswaran, S., and Pandya, S., "Unsteady Low Mach Preconditioning with Application to Rotorcraft Flows," in *18th AIAA Computational Fluid Dynamics Conference*, (Miami, FL), AIAA-2007-4473, June 25-28, 2007.

[77] Potsdam, M., Yeo, H., and Johnson, W., "Rotor Airloads Prediction Using Loose Aerodynamic/Structural Coupling," in *American Helicopter Society 60th Annual Forum Proceedings*, (Baltimore, MD), June 7-10, 2004.

[78] Renaud, T., D. M. O'Brien, J., Smith, M. J., and Potsdam, M., "Evaluation of Isolated Fuselage and Rotor-Fuselage Interaction Using CFD," *Journal of the American Helicopter Society*, vol. 53, no. 1, pp. 3–17, 2008.

[79] Sanchez-Rocha, M., Kirtas, M., and Menon, S., "Zonal Hybrid RANS-LES Method for Static and Oscillating Airfoils and Wings," in *44th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2006-1256, Jan. 9-12, 2006.

[80] Sanchez-Rocha, M., *Wall-Models for Large Eddy Simulation Based on a Generic Additive-Filter Formulation*. PhD thesis, Georgia Institute of Technology, 2008.

[81] Schlichting, H. and Gersten, K., *Boundary Layer Theory*. Springer, 8 ed., 2000.

[82] Schreck, S., "Two-Dimensional Unsteady Aerodynamic Processes in Low Frequency Shedding under Rotational Augmentation," in *47th AIAA Aerospace Sciences Meeting and Exhibit*, (Orlando, FL), AIAA-2009-1222, Jan. 5-8, 2009.

[83] Schreck, S., "Low Frequency Shedding Prompted by Three Dimensionality Under Rotational Augmentation," in *48th AIAA Aerospace Sciences Meeting and Exhibit*, (Orlando, FL), AIAA-2010-0640, Jan. 4-7, 2010.

[84] Schreck, S. and Robinson, M., "Blade Three-Dimensional Dynamic Stall Response to Wind Turbine Operating Conditions," *Journal of Solar Energy Engineering*, vol. 127, no. 4, pp. 488–495, 2005.

[85] Schreiber, O., *Aerodynamic Interactions Between Bodies in Relative Motion*. PhD thesis, Georgia Institute of Technology, 1990.

[86] Schweikhard, R., "Actuator Disk for Helicopter Rotors in an Unstructured Flow Solver," *Journal of the American Helicopter Society*, vol. 52, no. 1, pp. 58–68, 2007.

[87] Sezer-Udol, N. and Long, L. N., "3-D Time-Accurate CFD Simulations of Wind Turbine Rotor Flow Fields," in *44th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2006-0394, Jan. 9-12, 2006.

[88] Shelton, A. B., Braman, K., Smith, M. J., and Menon, S., "Improved Hybrid RANS-LES Turbulence Modeling for Rotorcraft," in *American Helicopter Society 62nd Annual Forum Proceedings*, (Phoenix, AZ), May 9-11, 2006.

[89] Shelton, A., *A Multi-resolution Discontinuous Galerkin Method for Unsteady Compressible Flows*. PhD thesis, Georgia Institute of Technology, 2008.

[90] Simms, D., Schreck, S., Hand, M., and Fingersh, L. J., "NREL Unsteady Aerodynamics Experiment in the NASA-Ames Wind Tunnel: A Comparison of Predictions to Measurements," Tech. Rep. NREL/TP-500-29494, National Renewable Energy Laboratory, Golden, CO, June 2001.

[91] Sitaraman, J., Katz, A., Jayaraman, B., Wissink, A., and Sankaran, V., "Evaluation of a Multi-Solver Paradigm for CFD using Unstructured and Adaptive Cartesian Grids," in *46th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2008-0660, Jan. 7-10, 2008.

[92] Smith, M. J, *A Fourth Order Euler/Navier-Stokes Prediction Method for the Aerodynamics and Aeroelasticity of Hovering Rotor Blades*. PhD thesis, Georgia Institute of Technology, 1994.

[93] Smith, M. J. and Potsdam, M. and Wong, T.-C. and Baeder, J. D. and Phanse, S., "Evaluation of CFD to Determine Two-Dimensional Airfoil Characteristics for Rotorcraft Applications," *Journal of the American Helicopter Society*, vol. 51, no. 1, pp. 70–79, 2006.

[94] Smith, M. J., Koukol, B. C. G., Quackenbush, T., and Wachspress, D., "Reverse- and Cross-Flow Aerodynamics for High-Advance Ratio Flight," in *35th European Rotorcraft Forum*, (Hamburg, Germany), Sept. 22-25, 2009.

[95] Son, J. and Hanratty, T. J., "Velocity Gradients at the Wall for Flow Around a Cylinder at Reynolds Numbers 5×10^3 to 10^5," *Journal of Fluid Mechanics*, vol. 35, no. 2, pp. 353–368, 1969.

[96] Sørensen, J. N. and Shen, W. Z., "Numerical Modeling of Wind Turbine Wakes," *Journal of Fluids Engineering*, vol. 124, no. 2, pp. 393–399, 2002.

[97] Sørensen, N. N., "CFD Modelling of Laminar-turbulent Transition for Airfoils and Rotors Using the γ-$\widetilde{\mathrm{Re}}_\theta$ Model," *Wind Energy*, vol. 12, no. 8, pp. 715–733, 2009.

[98] Spalart, P. and Allmaras, S., "A One-Equation Turbulence Model for Aerodynamic Flows," in *30th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-1992-0439, Jan. 6-9, 1992.

[99] Spalart, P. R., "Detached-Eddy Simulation," *Annual Review of Fluid Mechanics*, vol. 41, pp. 181–202, 2009.

[100] Spalart, P., "Topics in Detached-Eddy Simulation," in *Computational Fluid Dynamics 2004* (Groth, C. and Zingg, D.W., ed.), pp. 3–12, Springer Berlin Heidelberg, 2006.

[101] Speziale, C. G., "Turbulence Modeling for Time-Dependent RANS and VLES: A Review," *AIAA Journal*, vol. 36, no. 2, pp. 173–184, 1998.

[102] Standish, K. J. and van Dam, C. P., "Analysis of Blunt Trailing Edge Airfoils," in *41st Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2003-0353, Jan. 6-9, 2003.

[103] Stone, C. P., Tebo, S. M., and Duque, E. P. N., "Computational Fluid Dynamics of Flatback Airfoils for Wind Turbine Applications," in *44th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2006-0194, Jan. 9-12, 2006.

[104] Stone, C. and Barone, M. and Lynch, C. E. and Smith, M. J., "A Computational Study of the Aerodynamics and Aeroacoustics of a Flatback Airfoil Using Hybrid RANS-LES," in *47th AIAA Aerospace Sciences Meeting and Exhibit*, (Orlando, FL), AIAA-2009-0273, Jan. 5-8, 2009.

[105] Stone, C., Lynch, C.E., and Smith, M.J., "Hybrid RANS/LES Simulations of a Horizontal Axis Wind Turbine," in *48th AIAA Aerospace Sciences Meeting and Exhibit*, (Orlando, FL), AIAA-2010-0459, Jan. 4-7, 2010.

[106] Strelets, M., "Detached Eddy Simulation of Massively Separated Flows," in *39th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2001-0879, Jan. 8-11, 2001.

[107] Szydlowski, J. and Costes, M., "Simulation of Flow Around a NACA 0015 Airfoil for Static and Dynamic Stall Configurations using RANS and DES," in *AHS International 4th Decennial Specialists' Conference on Aeromechanics, San Francisco, CA*, Jan. 21-23, 2004.

[108] Thepvongs, S., Cook, J. R., Cesnik, C. E. S., and Smith, M. J., "Computational Aeroelasticity of Rotating Wings with Deformable Airfoils," in *American Helicopter Society 65th Annual Forum Proceedings*, (Grapevine, TX), May 27-29, 2009.

[109] Travin, A., Shur, M., Strelets, M., and Spalart, P. R., "Detached-Eddy Simulation Past a Circular Cylinder," *Flow, Turbulence and Combustion*, vol. 63, no. 1-4, pp. 293–313, 1999.

[110] Tung, C., Caradonna, F. X., Boxwell, D. A., and Johnson, W. R., "The Prediction of Transonic Flows on Advancing Rotors," in *American Helicopter Society 40th Annual Forum Proceedings*, (Arlington, VA), May 16-18, 1984.

[111] Vatsa, V. N. and Carpenter, M. H., "Higher Order Temporal Schemes with Error Controllers for Unsteady Navier-Stokes Equations," in *17th AIAA Computational Fluid Dynamics Conference*, (Toronto, Ontario, Canada), AIAA-2005-5245, June 6-9, 2005.

[112] Vatsa, V. N. and Lockard, D. P., "Assessment of Hybrid RANS/LES Turbulence Models for Aeroacoustics Applications," in *16th AIAA/CEAS Aeroacoustics Conference*, (Stockholm, Sweden), AIAA-2010-4001, June 7-9, 2010.

[113] Wilcox, D.C., *Turbulence Modeling for CFD*. CDW Industries, 2nd ed., 1998.

[114] Wissink, A., Sitaraman, J., Sankaran, V., Mavriplis, D., and Pulliam, T., "Multi-Code Python-Based Infrastructure for Overset CFD with Adaptive Cartesian Grids," in *46th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2008-0927, Jan. 7-10, 2008.

[115] Xu, G. and Sankar, L., "Development of Engineering Aerodynamics Models Using a Viscous Flow Methodology on the NREL Phase VI Rotor," *Wind Energy*, vol. 5, no. 2-3, pp. 171–183, 2002.

[116] Yu, Y. H., Tung, C., van der Wall, B., Pausder, H., Burley, C., Brooks, T., Beaumier, P., Delrieux, Y., Mercker, E., and Pengel, K., "The HART-II Test: Rotor Wakes and Aeroacoustics with Higher-Harmonic Pitch Control (HHC) Inputs - The Joint German/French/Dutch/US Project," in *American Helicopter Society 58th Annual Forum Proceedings*, (Montreal, Canada), June 11-13, 2002.

[117] Zahle, F., Johansen, J., Sørensen, N. N., and Graham, J. M. R., "Wind Turbine Rotor-Tower Interaction Using and Incompressible Overset Grid Method," in *45th AIAA Aerospace Sciences Meeting and Exhibit*, (Reno, NV), AIAA-2007-0425, Jan. 8-11, 2007.

[118] Zahle, F. and Sørensen, N. N., "Overset Flow Simulation on a Modern Wind Turbine," in *26th AIAA Applied Aerodynamics Conference*, (Honolulu, HI), AIAA-2008-6727, Aug. 18-21, 2008.

VITA

Charles Eric Lynch was born on Decemeber 5, 1981 in Atlanta, Georgia. He is married to Elizabeth Anna Whitaker Lynch.

Eric developed an interest in flying machines at an early age. During a summer program at Georgia Tech while he was still in elementary school, he obtained a course catalog and was immediately fascinated reading the descriptions of courses with names like "High Speed Aerodynamics." In 2000, he graduated from Lakeside High School in Atlanta. He earned a Bachelor of Science in Aerospace Engineering from Georgia Tech with highest honors in 2004 and a Master of Science in Aerospace Engineering in 2007. It was during an internship in the wind energy department at Sandia National Labs in Albuquerque, New Mexico in the summer of 2006, that Eric first became interested in using CFD to simulate wind turbine aeromechanics. In 2011, he completed a Doctorate of Philosophy in Aerospace Engineering at Georgia Tech.

Eric is a member of the American Institute of Aeronautics and Astronautics and the American Helicopter Society.